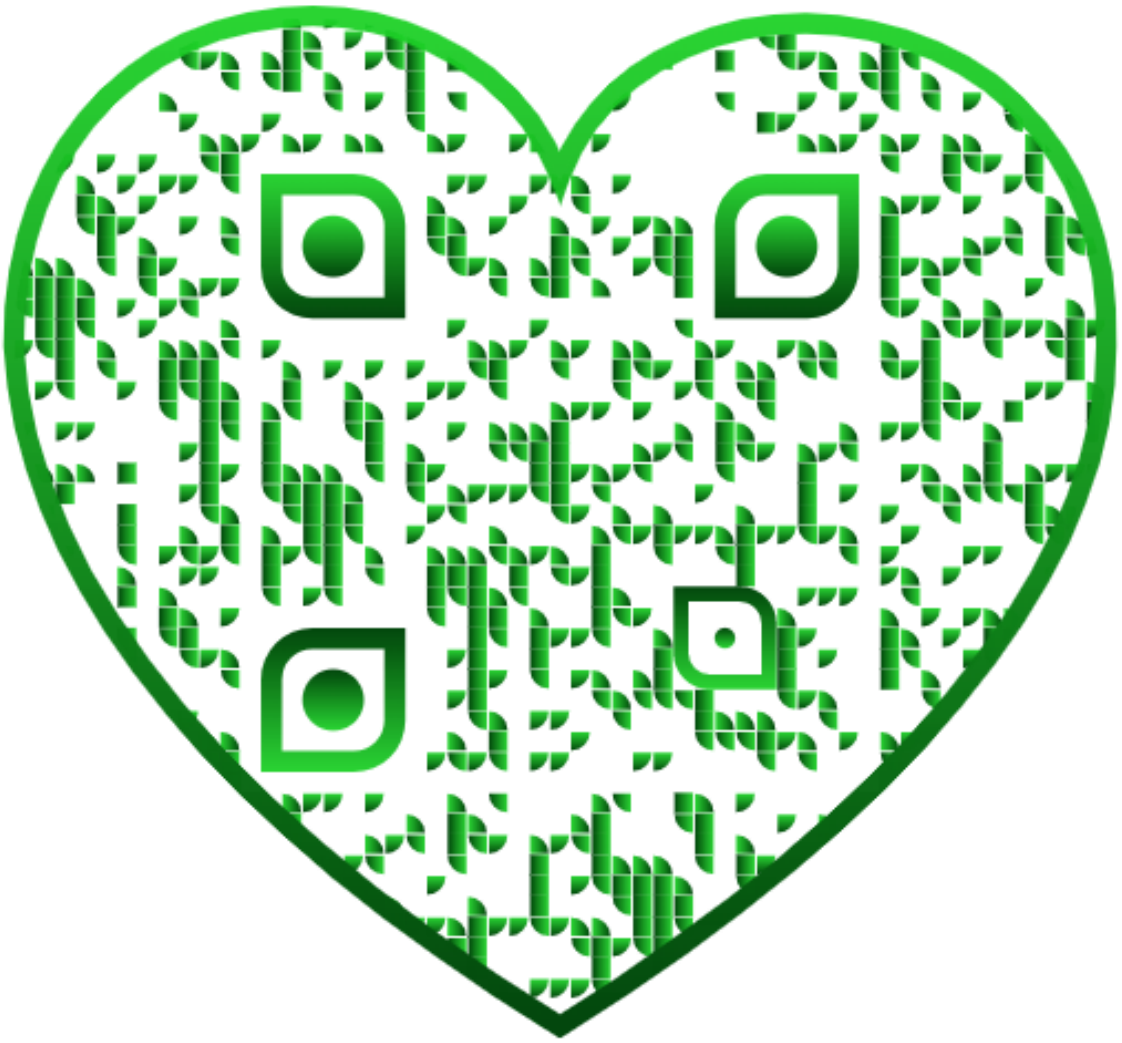
# THE ALGORITHM OF THE UNIVERSE

A new perspective to cognitive AI

Ancient Philosophy



# The algorithm of the universe

A new perspective to cognitive AI

## Author's Note

## Introduction

**Learning from the restrictions around us**

**Looking for cracks in façade of the created world**

**Why do I think the Sanskrit literature should be correct in its explanation?**

## Chapter 2: Why does our AI seem fake?

**The limitations of AI learning algorithms**

**What are we missing?**

- Learning a logic in a data rather than learning a skill
- Rigid, digital data representation
- Missing continuity in the algorithms
- Accurate computations is a necessity

**What do we need to create an AI that does not seem fake?**

## Chapter 3: Guiding principles of the algorithm

**The starting point (The Brahman)**

**Effective data representation**

**Inherent states of data**

- The indeterminate state before data (OM)
- The active state of data (devA or Energy)
- The dormant state of data (tamas)

**Data as primary driver of algorithm**

- The presence of a highly efficient, continuous, real-time foundation method
- The compose-decompose philosophy
- Fundamental architectural components
- Equilibrium as a abstract recursive goal

## Chapter 4: The algorithm of the internals of working of reality

**Starting at a dopant**

**A generic random data traversal technique**

**The controllable active-inactive cycle**

**Learning methodology**

**Learning with work intent**

**The work intent cycle**

- Primary energy with capability to recognize change (Mahadeva and Shakti)
- Logic to form a creator (Saraswati and Brahma)
- Resources to establish equilibrium (Lakshmi and Vishnu)

Forces of accumulation

**The progress of cycles (shAnti)**

**The work cycle (Exploding the controlled part)**

Vijnanam or coherent set of knowledge

Data persistence and the promise of a pattern (smara and pramANA)

The moving apex, the path and the quality of direction

The pattern, interference and work

Extension and expansion

**Tying it all together into an algorithm**

**A highly efficient real-time search algorithm**

## Chapter 5: How does this translate into this reality around us?

**Appearances and actuals**

What is cognition or awareness?

What is an individual?

**The fifth dimension**

Impact of actions in the rendering on the work intent

Evolving beyond the fifth dimension

**What is common sense? How is it established?**

**Space and time**

Distance and motion as we know it

A different view of scientific experimentation (energy, work, flow & fluidity)

## Appendix: Can we implement an AI algorithm using these concepts?

Trying to solve the problem of data representation

A learnable as a processor rather than a pre-defined instruction set processing

# Author's Note

There are many attempts to explain the existence of the universe. Even the observable universe around us is filled with unknowns and numerous unexplainable phenomena. Quantum physics attempts to explain it with physical particles such as The Higgs Boson particle and the Higgs field. There is an attempt to explain it using string theory. Yet, none of these theories can be replicated to show the formation of an universe nor do these have an explanation as to the why a specific path is followed by this universe to reach this current state. Chemistry attempts to explain the various reactions between atoms and molecules and form newer and newer materials. Yet, none of these materials formed nor the reactions have an explanation of the why the reactions occur. Mathematics is considered to be an universal language, yet, its constructs are too simple to represent any of the complexity in the universe and its operation. Wolfram puts this succinctly in his book "A new kind of science" and proposes a computation universe. It should also be realized that each and every theory is just trying to find an explanation for the existence of the observable universe around us, ignoring the fact that it is potentially possible that there is an invisible universe which reveals itself as we find newer ways to observe it. We need to first and foremost be able to abstract this universe as we observe it to be just a representation of something else underneath that is projected as this world around us by our descriptive thoughts. Then and only then can we find an explanation that is inclusive of everything that the universe is.

As far as we search for the origin of just these appearances of descriptive thoughts, we will only dabble with physics, chemistry, mathematics and computations. It should though be recognized that Wolfram introduces a very good concept. Simple things recurse to form complex worlds. But this without the correct transformation remains simply a recognizable pattern. How does this recognizable pattern go on to become this complex appearance of reality that we observe around us is up in the air.

The ancient Sanskrit literature has a very in-depth explanation of how this reality around us works and how it has come to be this. It just needs the correct context around which to translate these texts to understand it. This book is an attempt to bring out the algorithm that is described in these books to explain the existence of this reality around us.

For a long time I wondered what would be the best way to explain the algorithm of the universe as seen by the ancients and explained in the Sanskrit literature. I could explain it by using translation of the Sanskrit verses in a certain order. But I found that they do not take the reader into the journey that is necessary to understand and form the correct thoughts to understand the working of and creation of this reality around us. I thought I would write it as a description of astrology i.e., jyothiSha which, if we look at from the ancient view, is actually a science that tells us how to control the distribution of seeking which drives the formation of this reality around us and hence this reality itself. But, I find that the term has been highly mangled with all sorts of meaning associated with it. The mangling varies from predicting the future events in the form of astrology to getting naïve believers to fear the very meaning of certain terminologies that are used in this science. I find, even if there was a very beautiful science explained it becomes very difficult to break these pre-conceived perceptions and get the reader to read objectively to understand what is being said.

So, finally, I have decided the best way is to use the computers that we have created and the AI that we are trying to write as the best way to describe it because this was my primary driver to want to understand the Sanskrit literature and to debug the working of the universe. The underlying curiosity to understand of why and how the AI that we write does not even seem to be a "fake imitation" of the working of the universe around us. The underlying curiosity to understand how this reality around me is so completely immersive while the virtual reality that is created does not even come near this immersive experience that we have with reality. The curiosity to understand if it is even possible for us to create a hardware and software that comes even little bit close to the kind of intelligence, knowledge and beauty in the algorithm of this reality around us. This has been my driver to decode and understand the Sanskrit literature. So, I find that is the best way to describe it also. Hence this book is oriented towards this.

Do send me your ideas at ancientinsight1@gmail.com

# Introduction

Logic as a terminology, is commonly associated with the computer world. Applications, programs and other artifacts written in the computer industry is mostly oriented towards commercialization and hence falls under the applied logic category. It is mostly used in a sense to automate repetitive tasks that are considered a drudgery for the human mind to do without errors. But, it should be recognized that in no way are these, what humans consider as common mundane tasks, really common mundane tasks. For example data entry is considered a common mundane task, but, strangely, this cannot be automated with any accuracy easily. For humans, common household tasks such as cleaning dust, cooking and so on are mundane tasks. While we can create robots that do these tasks, they are limited to actions such as cleaning accessible surfaces and need human intervention to get it done accurately. Hence, we need to recognize that computers and logic only helps automate repetitive tasks that can be represented with the deterministic logic that we have identified and can code using the processors that we have developed.

AI is a loosely used terminology to indicate “so called intelligence programmed into applications”. The amount of intelligence created is limited to the intelligence of the architect of the program and the “domain” for which the intelligence is programmed. We can call it a learning algorithm. But, it should be recognized that even a learning algorithm is trained only to the extent of the input data. Further it is only trained to the extent of the algorithm coded by the programmer represented with weights and mathematical function representations that bind the various pieces of information in the data that are recognized already by the programmer based on an expert’s opinion. The common sense of humans which is termed as AGI or artificial general intelligence is typically not automatically present in computer programs and is also considered to be highly difficult to program. It should also be recognized that we have no way of even accurately defining what “common sense” really means. We can only define it by describing situations where humans apply common sense. When such is the case, how can we code in common sense into a program?

When we view reality, we always view it as physics of matter, mathematical equations, chemistry of particles, atoms or molecules or biology of the cell that forms a living organism. According to

modern science, logic has no place in the study of the observed reality. Logic is shelved to the world of virtual where some computer or processor follows a series of instructions to either reduce the work load of a human or for an entertainment experience by a human.

Yet, we find that, if we de-construct the working of any process in reality it just boils down to some sort of logic identified by us. For example we can express a logic when water is boiled as: “measure temperature of molecules. If molecule temperature excites molecule to have a force beyond the adhesive force between other molecules, then break the bonds. If bonds of molecules of water is weak then rise to surface. If molecule bonds can break surface tension, then break and rise into atmosphere. Surface molecules sink down to get heated.” And so on, at the minimum, a whole lot of rules (coded as if-else clauses) can be applied to understand the working of boiling of water, which is pure logic operating on an observable called water molecule.

We are forced to accept, albeit with a very high level of reluctance and resistance that the observable world around us is just a by-product of our brains creating “a world” based on both external and internal inputs which we are unable to perceive in the raw form. Many fail to even recognize or accept the truth of this. For example, while the input to the sense organ of the eye is light and translated to electrical signals, the output that we view is an image. Similarly while the input to the ears are sound waves translated to electrical signals we neither see the perceived waves nor the electrical signals, but have translated it as either music or words or higher level images formed by stringing together these words together. So, if these are not just a by-product of the brain, what else are they? A very revealing example can be seen in the binaural beats, where two known high frequency waves can be projected to our two ears and we translate it as a wave of the difference of the two wave frequencies.

Further, if, we accept that this is case, that what we construct of this world is just a translation of input signals, it needs to follow that each and every being within this reality can possibly have created a totally individual world very specific to the translative capacity of the brains they are equipped with. Even though a common rule can exist between the various beings, the worlds we are live in are distinctly different. As a proof, we just have to see the difference in reactions, descriptions and various other such parameters within human beings themselves to the same. Similar is the case with other living beings such as dogs, birds, insects and so on. Reactions to

same events vary between each and every being. The same environmental disaster invokes varied responses in different beings, going on to prove that the world that we observe and describe to ourselves need not be the same as the world that another being observes and describes to themselves. We are hence forced to accept that, the number of observed realities has to be as many as the number of beings present in this universe.

A pale imitation of this can be seen in the varied data analytics programs that are written. Data analytics takes input data either from IoT sensors or from various interactions of the user with some application and this data is scrubbed, bucketized and knowledge searched in it. This sort knowledge is then used to achieve various results such as showing related advertisements, selling of relevant products and so on. Given that every human exists in a world created by their own brains, it should also be recognized that, the algorithm written by that human, tends to "mimic the logic" in their world and hence acquires the same or more limitations as set in their world. It is rarely seen that two programmers can come up with the same algorithm to solve a problem unless they are both taught the solution and the trained solution is re-created instead of writing an original logic. Even then if the solution training is in the form of a description of logic, then all the way from the pseudocode to the program written varies for the same descriptive solution. This goes to reinforce the fact that both programmers live in different worlds, the worlds reflecting their observation capacities of this reality around them.

The data analysis is done by our thoughts on the properties detected from the truth and the result is this observable world around us. So, it should be recognized that none of this perceived world around us is the truth, but is just a palatable version of the truth given to us in a smoothed fashion rendered by the brain. Studying and changing anything in this world is just like trying to change one rendered UI for the same program that runs on a single server. Unless, the change that is rendered in the UI is not translated to the server program via API, the change is temporary and is lost once the UI is quit. Similarly, physics, chemistry and so on science that we have in the modern world is just a study of the specific rendering by a single being extended to an observation and rendering by other beings. It just modifies temporarily the rendered world around us rather than the underlying truth that renders this world. So, how do we understand and change the real truth that renders this world, if even it see and study the world all we can do is see is a rendering?

Hence, the Kena Upanishad as aptly asked the question:

*na tatra cakṣurgacchati na vāggacchati no manaḥ |*

*na vidmo na vijānīmo yathaitadanuśiṣyāt || 3 ||*

*Translates to:*

*Images cannot capture it, words cannot describe it, mind cannot grasp it nor can wisdom understand it. Then who or how can this be studied?*

As I have said so in many of my blogs and books, this is the core of all the problems that we have. How can the truth be studied if every time we try to study it all that happens is the formation of a thought which brings us back to square one where this rendered world is formed.

We also need to understand that we have no clue as to how our actions impact the underlying truth and in-fact if it impacts anything at all in that underlying truth. We tend to conduct experiments to prove our theory and call it a proof for a theory. But what should be recognized is that if the "brain" is translating and creating a world around us, then it is quite possible that conducted experiments and their outputs also have to confirm to the world that is created by our brains. It does not prove the theory one way or the other. It just goes to show that the illusion of the created around is completely and absolutely holistic and cannot be cheated in any way to reveal its secrets. This is beautifully explained in the Bhagavad Gita verse which all of us know with a totally different explanation:

*karmanyevadhikaraste ma phaleshu kadachana*

*ma karma-phalahe-turbhur ma te sangostvakarmani*

*Translates to:*

*Always the result is bound to work invoked by the order, advancing the bound result of the order further only binds that accompanying to non-order.*

What this is telling us is that while a "work done" in the underlying truth to cause an "order in the unordered truth" always causes a result, the reverse is not true. By changing the advanced result in the form of some imagined "work done" does not change the order in the underlying truth and it still stays in an unordered state. This in-fact is another question that is asked in the Kena Upanishad which says this:

*śrotrasya śrotraṃ manaso mano yad*

*vāco ha vācaṃ sa u prāṇasya prāṇaḥ |*

*cakṣuṣaścakṣuratimucya dhīrāḥ*

*pretyāsmāllokādamṛtā bhavanti ||2||*

*Translates to:*

*It is said by ear we hear, by mind we think, by speech we speak, by eyes we see and by breath there is spirit. I, embracing what in this world, thereafter imperishable, exists steadily*

This brings out the problem correctly. What can I embrace in this world that I have created which will get imprinted on that truth from which this world is created. If, as we saw previously anything acted upon on the result will not change the truth, then the question is what will? How can we experiment to know the truth? The experiments that we conduct are just on the result of the order and hence should not change the truth in anyway.

So, why is it important for us to know the underlying truth and the way this reality around us works? After all if everything is just a rendering and I can render what I chose to believe, then I should be able to render what I want and forever live in this illusion I have created around me? The point becomes more and more important when we start writing the AI algorithms. The reason these algorithms fall short every time is because our brains are highly sensitive and can detect fake very easily. While the brain deceives us into believing this world around us is the truth, it is very difficult for us to fool our brains into believing anything and everything to be the truth. To develop a true AI we need to be able to understand and mimic the steps that the truth has taken to create this model within us.

## Learning from the restrictions around us

The precision of measurements vary by the instrument used for measuring. For example, if we use the standard ruler that has a minimum measurement of a millimeter and a maximum of 32 cm, to measure length, then the precision of the length measured is only up to one millimeter, anything less cannot be measured. But, if we use a micrometer to measure we can get a precision of micro-meter. The technique used to measure also determines the accuracy with which measurements can be done. Thus, we find that, by understanding the limitations of the instrument that we are using to measure we can glimpse the data that we are missing out, even if we did miss out the data.

To understand the limitations of the measurement instrument, we need to understand what is the measurement instrument that we are looking at, when we study reality around us. To do this, rather than study of reality around us from intently experimenting with the external, we need to change the perspective to studying the relation of “ourselves” to the external, because the observation is done by us. When we do this, we realize that “this body(for now)”, can be considered as a measurement instrument. The question then arises, what is it measuring? The answer has to definitely be “some set of properties of that which is rendered as this reality around us”. That which renders as reality around us, we call as the “underlying truth”. These measured properties of that underlying truth are analyzed by us and conclusions drawn which forms “the individual world” of reality that we perceive and believe to be reality.

The limitation that we need to realize in this setup is that this measurement instrument called “the body(for now)” and the analytical output and conclusions drawn are based on the properties that can be measured. This is the capability of the being. Thus we can only know as much as we are capable of reading the truth. It should also be understood that a perception of a property can only occur once the truth has acquired the given property and there has been a change from some original value of that property. If a change does not occur or a potential change can occur, we do not have a way to perceive it.

So, how can we change this which is stacked against us, in knowing the underlying truth?

We need to recognize that while the limitations of the measurement instrument limits our perception of the underlying truth, studying the logic that we apply to understand the reality around us should provide us clues as to how our own analytical engines work. An understanding of this logic should help us reverse apply the logic to see the raw data on which this logic was applied to get at the conclusions drawn.

## Looking for cracks in façade of the created world

The modern world is built on the descriptive intelligence of our being. We need to realize that this is so integrated in us that we do not stop to realize the various individual steps involved to come to a conclusion or a final description that we perceive. For example, when we see “that it is raining”, though, we do know that it involves a path that goes through “the process of evaporation of water from

various sources on the ground, a collection of these into clouds and a precipitation of that water from the clouds", we rarely remember or observe that entire path when it rains. When it rains we just observe the consequences of rain such as a wet road, a flooded street, destruction of houses by a cyclone and so on.

It should also be recognized that when we search for proofs related to the reason for the "falling of the rain", that the "reasoning" also goes via the analytical engine that defines us as a being, and creates this façade world of reality. Thus, the experiments that we run, the reasoning that we come up with, the results that we observe and the description given to the observed results, all has to hold together to support the analytical analysis, output and conclusions reached by the analytical engine that forms us, as beings. Thus, using such a methodology of experimentation with the external world, we cannot find the raw data that is used in the analysis, the underlying truth from which the raw data was perceived to form the analytical output nor the logic or threads that leads to the analysis output and conclusions.

Hence, a conscious experiment of any nature cannot lead us to an understanding of the underlying algorithm nor the underlying truth. If we need to understand the underlying truth by observation of the external, we need to find cracks in the façade that the analytical engine has created. These need to be cracks where the data perceived falls through without any analysis performed on it.

The characteristics of a data that falls through cracks in the façade, is that all beings should perceive such a data without processing and observe it in its raw form. It should also be noted that such a data that falls through the cracks tend to be concepts that adopt the qualities of the observation rather than have an inherent quality of its own. For example, when we recognize that there is a concept called "space", "space" itself has no attributes or qualities associated with it and is "visible" due to the presence or absence of matter in it. Thus space has adopted the qualities of matter when perceived rather than stand out as an individual entity with characteristics. We find that "time" is also one such entity where "time" has adopted the quality of change in the subsequent "space frames" rather than stand out as an individual entity by itself. Hence, any being that perceives "space and time" irrespective of what type of being they are, follow the standard principles of adopting the perceived qualities of the contained matter or change in matter in "space" and "time" respectively. Thus, pondering on what space and

time is should allow us to look through a crack in the façade by the analytical engine.

The next aspect to look at to find such cracks is related to the working of the analytical engine itself. As, I had indicated in my book Surya Siddanta: Emergence of empirical reality, typically the data that is analyzed for which an analytical output is given is that data which is perceived to be closer in distance to us as a being. As the distance radiates out, with us as the center, we find that we do not analyze it because there is not enough data to analyze. Typical analysis is done only for data that is considered to be something that will affect our existence as a being. This means that when we look outside the earth itself and look at the stars and the planets we should not be analyzing the data more than the minimum of fitting the observation into our known concepts. Hence, these can also be used as cracks in the façade to understand the underlying truth.

It is sad that we try to break this inherent barrier to our analytical engine by adding external perceivers such as sending satellites and drones to these planets and stars to study just the observable properties, rather than using these barriers to enable us to see beyond the analytical engine and understand the logic of the paths of the truth or the raw data that drives that logic. The Sanskrit literature has done just this. Many of the Sanskrit works attributed to astronomy and astrology have studied these external elements to enable the inherent barrier to the analytical engine act as an enabler to view beyond the analytical engine that creates the description of the concepts.

## Why do I think the Sanskrit literature should be correct in its explanation?

So, why do I think that what is explained in the Sanskrit literature should be more closer to the truth rather than what we have found using what we call “science”? The explanation is best given by comparing the evolution of reality around us to the evolution of various technologies that we develop. “Evolution” is another concept in modern science that has been applied without any context, and applied only to just the observed matter and forms that we observe around us using external senses. Hence it has become difficult to envision it when the question arises, “How did an ape suddenly by itself start walking on two legs, expand its brain to contain a frontal lobe that can do analytical thinking and in the process wipe out a whole lot of other beings. It becomes difficult to

imagine the probability of all the required elements coming together to start off the formation of what is called a living being within the short time that we see as the origin of the universe." Hence a number of objections are raised to the theory of evolution that is alternated with the creationism theory.

It is strange that when we think of evolution, we always only see the body or the matter and think of it as the only one that is evolving, rather than the whole being, which is everything from knowledge, intelligence, awareness, consciousness along with the body. We just don't seem to want to account for any of the hidden elements at all. We seem to believe that all these automatically developed by the right mix of matter, which is where we are made our first mistake. Hence, we have the creationists trying to say that these were created by some external being called "God" or in some cases an alien, without taking into consideration that even if an external being created all this, how did that external being evolve or get created?

If we take an example of AI's evolution, we find that we are trying to develop the algorithm first before giving it a body in the form of a robot. As I have explained before, we are always limited by the instrument that we use to measure. But, that is not all. Our logic should also be limited by what we are and what we have embedded within us. It has to be an indication of how we evolved or how we are built. We are not capable of logic beyond what we use to live ourselves. We need to start considering the fact that evolution did not start at material, light or any of the observables that we can observe in this reality around us, but at a totally different unimaginable point of some unmanifested, unordered indeterminate truth, as I have explained later on.

To understand this, we just need to pick up the latest technology that has evolved namely the computer industry. Anything related to computer industry was not present and that is where we have to start. The computer industry itself should have started with formation of the hardware that allowed us to do simple computations such as the calculator. From there it progressed to a more sophisticated processor that could handle complicated data. During all those times, we had to understand the underlying working of the hardware to write any sort of meaningful program. From here the operating system got written which freed the programmers to write program without understanding the hardware, but had to understand the OS concepts. And so on it has progressed, till when modern programmers have no clue about the hardware, the OS, the

compilers and many such internal concepts that has led to the scripting languages which make it very easy to write a piece of code. It has progressed now to a point where the UI or the rendering for the backend code is given prominence and importance. The internals of the computer is a knowledge that is right now restricted to a very few programmers, where as when the computer industry began, all computer programmers knew it and used it. But, the UI programming itself was something which at the beginning given the least precedence. We can see similar is the case even with electronics. Evolution does not start on the outside and go internal, but starts evolving at the internals and adding step upon step over it, it spreads out to the external. So, why do we expect any different from nature and the evolution of beings?

We need to start asking ourselves if reality is not the same kind of evolution. Is it not possible that the underlying internals should have been the first to evolve and then the materialistic world evolved? Which implies that in whatever “form” our evolutionary ancestors may have had, they definitely should have had much better understanding of the internals of the working of reality, because this was necessary for evolution. I believe it goes to explain the reason why spoken language evolved before written language. It also goes to explain why the Vedas and Upanishads were infact transmitted by word of mouth rather than by written language, because that is the evolutionary path that got followed, description by thought, then mouth, then written language.

So, yes, I believe the Sanskrit literature must be more or less accurate when it comes to describing the internals of the working of reality. It also goes to explain why none of the Sanskrit literature talk too much about the material world and matter. It neither talks about our modern science such as physics nor of mathematics or any such science as it is claimed. It is purely the evolutionary path that has been traversed to get to this point in reality.

There was a time and I have mentioned this in my previous books, where I wondered “how is it possible that the ancients were able to hold their thoughts so steady that they are not mesmerized by this object world around us and are able to see the deep internals?” But, now it becomes clear, because they have traversed the path without the completely evolved fifth dimension to distract. Hence, it is simply a matter of documenting that which was the path at any given point in time. Just as how the concepts of OS evolved without the distraction of easy programming of scripts, but the easy

programming of the scripts were a vision that the programmers of OS wanted to achieve.

So, rather than seeing evolution of “matter” and transformation of one classification of being into another classification, if, evolution theory had looked at the evolution of just a “human” from that unmanifested, nonexistent to what they are now, then the path would have been different from what is suggested of it evolving via fish and apes. The resemblance to fish and apes would just have been an adaptation of thoughts, rather than hierarchical.

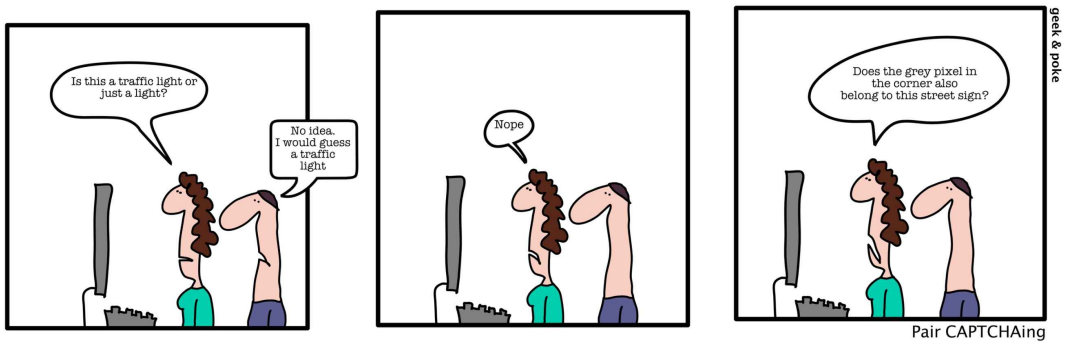
In this book I explain via the verses in Surya Siddanta and the BHPS, the algorithms proposed in these literatures for the working of this reality around us. I also compare these concepts to what we have currently present in computers and explain how they are different from what is explained in these books and why the concepts we have in computers and the way they are implemented, are not sufficient to create the cognitive AI that we want to create. Further, I try to put down my thoughts on how or what can be done to create a system that can be used to implement a cognitive AI system.

# Why does our AI seem fake?

In the current world of computer science the term “machine learning” has been used in varied sense. Learning is a loosely used term that is applied to many types of algorithms. The learning itself can be “learning weights and bias to be associated with already form fitted mathematical functions” using linear regression, logical regression, multivariate regression. Or “learning a smoothening function for a bunch of data points” using Bayesian algorithms. Or “learning how to cluster data points with weights” using algorithms such as k-means clustering. Or “learning how to classify data points” using the k-nearest neighbor algorithms. Or various neural network based algorithms where states are represented as mathematical functions learnt as a part of a multi-depth neural nodes allowing modified input data to pass through and hence achieving deep-learning. Many other such small and big algorithms are termed as learning. Further the types of learning vary anywhere from supervised learning to semi-supervised to unsupervised to reinforced learning. In a supervised learning the input and output data are forced on the algorithm to ensure that the learning algorithm adjusts itself to the expected output, while in a unsupervised the input data is learnt from and the output not controlled. In a reinforced learning the output is rewarded or penalized to adjust to the required values appropriately. But, in all of these type of learnings, we have moulded the output like a clay toy to give a certain output that is inline with what we or an expert in a domain thinks of as the intelligence required of the algorithm. But, is that true intelligence? Can intelligence be really judged to be correct or wrong? Can there be a wrong intelligence?

We live in a world where these algorithms have surprised us at every step. Even humans cannot achieve a few things these AI algorithms achieve. For example, many studies have been done where it is determined that an AI algorithm can very easily fool the captcha algorithms that is used to determine if a human is using the system or if a computer robot is using the system. I find that the more difficult the captcha is made, the more easy it seems for an AI to pass rather than the human. Humans are equipped with ever degrading sense organs such as eyes and ears to clearly identify unclear alphabets. Moreover, with every variation in the letter rendering or picture rendering, the approximation modelling present in humans gives back a number of matching scenarios, that choosing

the best match is just a matter of opinion anymore, rather than the correct solution. The simple mathematical computations are best done with computers than human mind which is the whole point of inventing computers. I would tend to also say, a computer generated picture or sound is best understood by a computer rather than a human/ Using it to determine if someone accessing it is a human or not is pretty foolish when a computer can do a better job of it. If we did use it, possibly we need to validate for a negative scenario. Verifying if a human mistake is made, rather than get it correct. The below cartoon in geek-and-poke says it as it is:


Is this a traffic light or just a light?
No idea. I would guess a traffic light
Nope
Does the grey pixel in the corner also belong to this street sign?
geek & poke
Pair CAPTCHAing

Another example where these algorithms have surprised us, is the storing of user's browsing history and using it to provide suggestions in terms of advertising for various products to be bought or actions to be taken. Sure, a computer is better at this than any human. We humans are definitely not very good at trying to track each and every action of another person and act according to that person's interests. We always only do it best when it is in our own interest. While we can say that a computer devoid of emotions is better at looking at browsing history objectively to spit out advertisements, it is very doubtful if I agree that it does a good job of it. When we look at the various suggestions and advertisements that pop up while browsing, it does make us smile or get irritated based on what is being advertised and our mood at that given point in time. Why would I want to buy a washing machine twice in a row or a fridge twice in a row even if they are of different brands. Sadly this is what is advertised. The best is when amazon suggests me to buy my redmi phone along with a carbon phone and a gorilla glass for the carbon phone. Seriously, why would I buy two phones together? While it is uncanny as to how the system suggests what to buy or not, the algorithm is purely permutations and combinations of words without too much of meaning associated with it. So, while the algorithms themselves seem to be very intelligent in knowing what I

have done, it is only because we do not expect it of the computer programs that it surprises us.

Yet another example of such algorithms can be seen in the voice processing systems that are integrated into the google maps and directions. Again it is uncanny to see the number of suggestions that are auto suggested by google as you are travelling. But, many a funny episodes have also occurred when we see old people or children interact with the voice system under the assumption that they are as intelligent as a human. The woeful lack of understanding of even the basics of a language as a colloquial version, is visible when we see a voice based system try to interpret the various different ways in which the same intent can be conveyed. When we talk, we rarely try to be grammatically correct or even syntactically correct by pausing and sighing at various words. We only try to convey meanings to the listener. But, this is definitely not understood by the systems, because they are coded for the formal language usage. These systems only look for key words or are coded for a specific sequence or grammar or some standard pronunciation of the words that any variations from that give rise to the funny interpretations and situations.

Yet another example is the various virtual reality games that try to mimic real life. While they are immersive and addictive in nature, it should be understood that it is still only the addiction that gets us to fool ourselves into thinking them to be real or immersive.

Humans are highly adept at remembering that which is pertinent to their current situation. From a humungous expanse of disparate data that has been collected over our lifespan we can recollect the appropriate required information that applies to the situation we are facing to get ourselves out of or react to the situation. This is like the cocktail party effect. We can easily attend a party and ignore all small talk around us until someone mentions our name. Then the whole focus changes to the person talking about us and we are able to easily switch the conversation to the relevant topic. Small irrelevant words that makes no sense out of context, actually makes sense to us based on the person talking about it and we are able to pickup and continue a chat as if it were a middle of some conversation started much earlier.

But, “a focused clear-cut collection” related to a specific search topic is purely a computer algorithm. As humans we can neither be limited to a single topic nor can we remember all the details of the single event. We always tend to associate it with our impressions and

it is the impressions that trigger the recollection of the details of the event. So while a designer-designed “Turing test” can be applied to a focused area of intelligence and the computer can be made to randomize appropriately between what is considered to be right and wrong to pass the test and successfully to imitate a human, whether the test will really be sufficient to say whether we are interacting with a natural being or not is highly debatable. It is like the saying “You can fool some of the people some of the time, you can fool all of the people some of the time, but you cannot fool all of the people all of the time”. It just needs time for a person to adjust to the reality of the intelligence learnt by a computer to be able to pin-point that it is a computer and not human they are talking to.

To even start thinking of creating a truly intelligent system, we need to ask ourselves “Why does our AI seem fake to us?” How does the brain know that we have mimicked an intelligence and it is not the same as the intelligence created by this natural world around us? How do we know that it is a virtual simulation? To understand this we need to understand the in-depth working of reality and start understanding the difference between the way reality works and the way our AI algorithms work.

## The limitations of AI learning algorithms

There are dime and dozen websites that explain artificial intelligence, machine learning, classifiers, clustering and many standard algorithms that are classified as learning algorithms. So, I will not go into detail explaining how they all work here, but only touch on the limitations of the learning algorithms that we have.

As I have said, in this book I have purely focused on comparing neural networks and the related learning algorithms against the algorithm present in the internals of the working of reality. Mostly because, in my view all other algorithms does not even come close to being a learning algorithm, let alone being a “truly AI algorithm”. They are just mathematical functions and computations. Sure, they do learn the related weights, bias and many such parameters. But, that is not knowledge or intelligence, that is purely “a search for the most suitable set of constants that fit the given discrete data points to achieve the goal aimed for”. A smoothening curve-fitting algorithm is hardly a AI. It is purely an algorithm that decides on the right mathematical function that fits the maximum number of discrete data points input, ignoring others as outliers. Clustering algorithms could have come closer to having knowledge and intelligence, had

the algorithm been different. But then, most clustering algorithms also use weights and biases associated to node values to determine some form of mathematical distance between node values. Then based on these distances the nodes are clustered or grouped. Hence, these also fall under the category of just mathematical computation that groups based on the best possible group of values, hardly can be considered as AI. Yes a very specific focused intelligence, but then the intelligence is in the logic that has been coded rather than the logic being learnt. The "so-called learning" in any of these algorithms is very limited and the "knowledge learnt" has some pre-defined meaning, based on which logic is applied.

Neural networks are algorithms designed to try and mimic the neurons of the brain. Multiple nodes called neurons are created in the learning phase with each neuron associated with a propagation or activation function. The inputs to the neurons are weighed down by weights and the inputs biased by some bias. The weights, bias and activation functions can be learnt during the training phase. Layers containing many neurons are then created and connected into different types of graphs. The layers and the graphs also learnt during the learning phase. Based on the weights, bias and activation function of the neurons, the logic traverses to the next layer of neurons and controls the input that traverses to the next layer. Thus traversing via various neurons in the different layers of the network an output is generated. This can be seen as just a glorified, gigantic mass of multi-level if-else clauses. The "if-else" driven by weights, bias and activation functions instead of a binary '1' or '0'. The if-else is learnt based on the input data training given to the neural network.

In my view, neural networks also tend towards a very limited "knowledge learnt" and has some pre-defined meaning that is coded into the logic written. We can call what is learnt as neurons or give it any other fancy names. They are still weights of input values, biases associated with some summation of the input values or activation mathematical functions. Thus, the knowledge learnt is limited. But, neural networks atleast learn varying levels of "if-else clauses" which comes closer to *one form* of intelligence. When the number of nested levels is high, even an if-else clause can achieve seemingly intelligent decisions, because the combination of paths present increases. But, again it should be understood that it is still just a finite number of paths followed. It seems complex to us as humans, because our mind boggles at that level of nesting and the combinations of paths possible. Typically, when it comes to that

level of complexity, humans tend to usually reduce down the complexity and use a different algorithm. I find that the fact that the neural network does not optimize it down at those levels itself is indicative of it not being intelligent, but being just a tightly coupled logic coded by a machine rather than by hand.

So, what do I mean by “knowledge learnt” is very limited and has pre-defined meanings. Let us take an example. Let’s take for example a dog. I will tell the story of a litter of puppies that were born to a street dog born near my home. My front gate has a hinge that leaves a bigger space between the wall and the hinge. The space is enough to allow a puppy to easily squeeze into the portico area. We typically leave our footwear outside at this location for many useful reasons, let’s not go there. As all of us know dogs and puppies, especially the growing ones, like to chew on slippers. So, it happened, one day I found that my slipper was dragged outside the house and chewed on completely. Obviously ruining it. Now, I had to prevent the puppy from getting in. I did not want to invest in a permanent solution of any sort, so I like a fool I went to pit my intelligence against the intelligence of a natural instinct of the puppy. I have few potted plants growing in the front of my house. So, the easiest solution that occurred to me was to just pull out these pots in front of the gap so that they block the access to the gaps. Imagine my surprise when I found that the puppies were able to still get in. So, I watched to find out how they got in. I found that given that the pot had a narrower bottom than the top, they just squeezed in the gap between the narrow base and the gate and were able to do the gymnastics to get to the gap in the gate. Now, I said, that is bad. So I got a few bricks and blocked the bottom of the gap using these bricks. Yet again they were able to get in. I found they just jumped over the brick. They would just squeeze through the gap of the pot, jump over the brick by using their front paws to hold onto the top and get in. I tried with two bricks stacked one over the other. To my surprise, they just would push down the top brick and get in. It went on in this manner, till I found a solution that just prevented them from either pushing and getting in or squeezing in through some open gap that I had left. Took me sometime to get to the solution.

Let’s leave the part of my bruised ego and grudging admiration and appreciation for life’s intelligence. But, the purpose of this story is to show that “the same knowledge” can generate “multiple intelligence”. This means that knowledge needs to have the inherent capability to adapt to varying situations. It needs to be elastic. If we look at the story I said, the dog had only two pieces of knowledge.

To understand the space into which it can squeeze or crawl into and to understand its ability to jump over obstacles. It knows exactly when to use what to achieve its goal and that is intelligence. The admirability of the way the puppy was able to apply the knowledge shows how the knowledge can vary continuously in all directions without any restriction.

In neural networks that we build, we just vary weights associated with inputs and biases of some summation of the inputs to give prominence to one input over the other. One neuron can only be used rigidly for some matching inputs or a range of matching inputs based on how it is coded. It cannot be applied to what is called “relevant situation”. A “relevant situation” needs to have more neurons in the if-else clause. So, the variability achieved by that single node that stores the logic is very rigid having only a limited match to which it can apply. Hence, I say neural networks have very limited learning capability.

## What are we missing?

The deep learning algorithms are touted as algorithms based on how the neurons in our brains work and can learn a whole lot of knowledge. So, why don’t these mimic reality and how can we recognize these to be fake easily?

The simple fact is that we approach AI as any other experiment in physics or chemistry, where a given problem is taken and stripped down to a controlled environment and the concept tested. Subsequently just the concept is expanded to be applied as a technology to achieve some set of controlled functions. For example, we conduct an experiment on a specific material such as copper to understand that it can conduct electricity or a specific material such as silicon to understand that it can be used as a semi-conductor, subsequently we apply the experimented and proven principle to create a conduction wire for electricity or a transistor which are for very specific applications of the concept. Silicon as such can be a semi-conductor or emit photoelectrons that can be used to capture light images. But this concept was discovered later after semi-conductor was used to create a transistor. We tend to step very carefully, experimenting at every step first describing a concept, proving to ourselves that it can be repeatable and then subsequently applying the specific or a combination of known concepts to create a technology. The aim of such a type of study is to be able to describe using the language that we have defined, what is happening and to

be able to use the concept for achieving other purposes using some form of intelligence or logic only we can understand.

Knowledge in nature does not work in that manner. Nature, we find always has the entire continuity present. The whole gamut is present in nature, it is just our observation that limits the knowledge. We can see it in all the ways we apply natural knowledge or intelligence. We should not confuse application of knowledge to description of the knowledge. What we do in science is describe the knowledge, while in normal day-to-day life we actually apply the knowledge without describing and sometimes not even knowing how we are doing it. I do not believe everyone needs to know how and why we can walk, but we walk. I do not believe anyone needs to know how and why we can hear, talk, see, sense touch. But, we can sense and create images, listen to songs and many such tasks. Nature operates on reducing from a continuous whole to a reduced applicable whole, while in experiments we create a small reduced part and try to pick that and expand it to a continuous whole. While this can give results in some cases, in the field of knowledge and intelligence it introduces many problems.

So, why cannot we reduce the amount of knowledge and try to create intelligence out of it? This is because, we have no clue as to what knowledge goes to making up a certain intelligence. For example, we believe that we have decoded the working of our sight by understanding the way our eyes work. Our explanation says that light gets focused by the lens in ours eyes and falls on the rods and cones of the retina which then gets converted to electrical signals and processed by the brain to form the image. The question now is, if a camera is modelled on this concept, then why does not a picture capture all the aspects of the image that is created by our brains? Definitely everyone knows that seeing something actually using the eyes is not the same as seeing a camera captured picture. This can best be realized when we see the photograph of the grand canyon vs when we are actually standing at the precipice of the Grand Canyon’s lip looking down into the depths. Even if we forget the associated experience, just the image formed in our brains of actually seeing has much more details than a picture. Why is this so? The only explanation is that we have not understood all the concepts involved in formation of the image in the brain. Sure, we have created many technologies such as IMAX, 3D movies with a flying plane going close to the Grand Canyon depths. But, yet the real experience and image formed is much different. We can very easily say that the image in our brains is affected by the experience of all

the other sense organs. Isn't that what the IMAX and 3D or 4D movies are trying to provide? While it does give the thrill and experience, it is still different from the surreal experience of seeing the Grand Canyon. So, the question is it only that much? But, even without answering the question, just trying to recreate the base description of the working of the eye using a camera and it not sufficing supports the theory that we cannot reduce knowledge down into a controlled environment. Obviously we found that experience makes a difference to the way image is created.

In the current AI systems, it does not matter whether it is deep learning, machine learning, neural networks or any such technology that we create, we have reduced down and limited the scope of visibility of data available to be learnt from to a very small part of even the domain that is being learnt about, let alone other related domains. For example, if we wanted to create routes for transportation of goods from one place to another, we could just look at the distance travelled, cost and many such apparent parameters to create the route. But, typically when a human creates from experience, many parameters go into creating the route. Is the route going through a forest area? Is the route having good roads? How many twists and turns need to be taken to go through a route and many many such parameters that make sense only to us in the decision making.

For example, while the shortest route could be through city roads, when travelling from one location to another, we just may prefer to take a tolled road that is more than double the distance and costlier option between the locations. Such a decision can be based on the fact that tolled roads typically has no signals, no turns, has higher speed limits and with lesser interruptions which allow us to reach in half the time it takes to travel the shorter distance via city roads. Just the wait times at signals may take more than the time to travel the longer distance. But many a time, we also base the decision on the time and day of travel to take the decision. Thus, we see that restricting the amount of data available to the AI based on some expert's opinion just ties the algorithm to a single person's impression or understanding of the problem and there in starts the whole problem of our AI.

This lack of consideration of all the information that makes a difference in the decision taken appears as a lack of common sense, where we see obvious decisions not taken. This is one of the many factors that make us realize that the intelligence is from a coded system which cannot think for itself, and is just picking a single

possibility from one set of permutations and combinations available in a million other permutation and combinations available.

Let us take another very simple example. Based on the experience we have of reading many emails over a span of many years, we can very easily manually identify when an email is a spam. Say we wanted to create a program that recognizes spam messages in the email. We can hand-code a logic or create a learning system. When we hand code it, we code it as a rule based system that runs rules that a user sets up to mark messages as spam. Here, we have let the user use their experience to specify how a spam is recognized. What will the user do? They would look at a message that they think is spam, recognize word occurrences in the subject, word occurrences in the message of the email body, the from address and so on many parameters in the email and setup rules that match the observations made by the user. The system now runs these rules as an if-else clause when messages arrive and classifies them as spam.

Now, let's say we want to create a learning algorithm based spam detector. What do we do? We still create it as an if-else clause, because there is no other way in computers to recognize this. We can either ask users to mark a spam message as and when they see an email that is spam in their inbox. As many people keep marking spam, we have created an algorithm that has taken the common parameters of the emails such as subject, message body, from address and so on and based on the recognized commonality between the messages, we learn a set of rules to run, to mark the message a spam. In such a case obviously we can take the input and experience of many users into consideration to mark spam, which can improve the effectiveness. We could have coded in patterns of occurrences of various words, checked for presence of certain words in certain format, we could have added distance between words and many such logic. But, they all translate to the same algorithm. A set of rules, either hand coded or machine learnt or assisted machine learnt. So, what is the difference between what we hand-coded and that which is a learning algorithm? It is the same if-else clause, one hand-coded, one machine coded.

Thus, for us a learning algorithm, is just a way to "*machine code those scenarios which we cannot think upfront*". Rather than create real knowledge from data and use that knowledge to create intelligence to understand what spam email really means and hence classify spam emails, we are just using "learning algorithm" as a way to machine code the same algorithm we would have hand-

coded. In-fact, we do not recognize "*knowledge* present in data" different from *intelligence* at all. For our AI knowledge and intelligence are the same. The program that is created from such assumptions is obviously just another computing program rather than being a truly intelligent system. So, is the algorithm truly a learning algorithm at all?

**Learning a logic in a data rather than learning a skill**

When we look at how reality works, we find that living beings can start living without much training. They train on the job, so to say. Very small amount of data gives huge learnings. So, how is it that even with terabytes and terabytes of data that we have, we cannot train in a logic that is no-better than a gigantic if-else clause? The if-else weighed down by weights and biases, rewards and penalties which are pre-coded by so-called learning algorithms?

We may call it a learning algorithm. But, we need to accept that what is present is just another logic that is coded in the form of learning algorithm which auto-codes *rather than creating a skill*. This is just another way of "coding the if-else program". It does not matter if it is hand coded in the form of actual "if-else code" or "goto" or via "look ups in hashmaps" or learnt to be in the form of weighted input to nodes that run activation functions in multiple layers. They are all pre-coded if-else clauses in multiple forms which run on some data that is input to give an output. Why should it yield a different result as opposed to a hand coded algorithm which is coded specifically for a given data type?

Let us an example. The market which every auto-company is trying to capture, autonomous car or a self-driving car. Let us forget about a learning system that is being added to the car and think about how we would have coded the system if we were able to fix all the parameters and conditions of driving as the contours of the road, the inclines and declines, the road being empty and so on and had to hand-code it. Looking at this, much reduced problem, we find that, if we could access the car controls and accelerate the car to a required speed such as 80kms/hr, and left the steering wheel so that the wheels are oriented parallel to the curb, it should have followed the road contours if the road had been banked correctly for turns. Subsequently, we just had to accelerate and decelerate for the inclines and declines to maintain the speed of the car. But, given that roads in the real world cannot be banked so accurately, we will have

to adjust the steering wheel to adjust the car wheels as it veers off the course by some recognizable deviation.

To code this, we would have created a space representation of the 3D space with the road contours and all other features on the road sides coded into the 3D space object. We would have followed the current trajectory of travel of the car in that 3D space w.r.t the road contour. Project the trajectory of the car if it travelled at the given wheel orientation and speed. Computed the change required in the wheel orientation to follow the road contour if there is an non-acceptable deviation. If a change is required to the trajectory, we would have changed the steering wheel by the required amount so the trajectory of the car matches the road contour for the next 10s (we would have made this configurable). If there is a decrease in speed of the car, we would have computed acceleration required and if there is a increase in speed of car computed the deceleration required. Controlled the car accelerator to change it as computed. This we would have just been repeated till it reached its destination.

Now, if we take off the controlled parameters that we had put such as an empty road, known road contour etc., how would the program change? Since, we do not know the road contours, we will have to detect the road contours. Since we do not know the inclines and declines, we need to detect it. Since the road cannot be empty, we need to detect the presence of obstacles in the path. If we added the sensors for these, the same logic can work, except in the case of road contour rather than comparing it with existing contours, we need to compare it against the detected road contours for the next few minutes. Obviously, we need to have detected the road contour atleast a few minutes before to be able to act in real-time. The other change required would be to detect obstacles in the path and take appropriate action, go around, slow down to match speed and so on.

Obviously this is a humungous hand-coded logic that keeps changing with every situation we need to code in. We introduce traffic lights, we add a sensor for detecting red-light vs green light and code in the logic, we introduce junctions, we add sensors for detecting cross-traffic, we code in the logic to compute car trajectories and intersecting trajectories and take decisions, we add in left and right turns, we add in sensors to detect when the turn is coming and add in the trajectory calculation to change the steering wheel correctly. But, even if it is humungous, the fact remains that the logic can be hand-coded correctly if the sensors had been the correct ones and the number of resources available to hand code were also as humungous. The obvious problems with this is that,

with the number of situations and parameters we add in, we find that more and more logic needs to be hand-coded to handle the situation. And we all agree that the number of situations that can occur when we drive a car is nearer to infinite rather than finite and all scenarios cannot be hand-coded easily. It will make the code unwieldy and really complex to even start understanding, let alone maintaining, debugging and fixing bugs in it.

Hence comes the situation where we want the car to learn on its own *how to drive*. But, *what we seem to have is a learning algorithm that learns what we are hand-coding and take decisions*. The logic need not be what I have described previously, but some machine coded logic based on the data. This is typically what is done with the current neural networks. A set of scenarios are learnt by the neural networks based on some training data that has been fed into the system possibly by a human driving the car and feeding the input to the algorithm and / or feeding road details into the algorithm and many such parameters being fed into the system. But, the fact remains that this is also just "machine coding" the complexity into the network and letting that replay itself for actual situations.

This is not truly a learning algorithm for "*learning to drive a car*", this is a "*learning algorithm for the logic that is present in the training data and coding that logic into an network*". All this does is controls and takes decisions that we have taken when we created the training data by driving the car. This is where the problem lies. *We need a learning algorithm to learn to drive a car, not to recognize and learn to code the rules of driving. The rules of driving is a secondary knowledge to the primary knowledge of driving.*

What will be a truly intelligent system is not such a system with a learning algorithm that is pre-trained for decision making and controlling a car in given set of X situations based on some driving rules present in the learning data. This can possibly work in countries where drivers tend to follow the lane system and road ethics to the T, hopefully, and the traffic is streamlined within lanes and so on. But, for example in India, drivers do not follow lane systems or have any streamlined way of driving. Many a time, drivers do not even indicate they are turning, yet we do know they intend to turn and slow down for them. This kind of driving pattern can be driven by many factors such as the lack of infrastructure or the lack of space for the volume of cars or plain simply apathy of drivers, as in Bangalore. But, the fact is that a well-coded driving pattern learnt from any training data, cannot work in such a haphazard traffic condition. Here there is no situation which is

similar to any other situation. All are unique by themselves and we need to take decision based on real time data. The only rule here is "*there are no rules*". This is the situation where skill is absolutely necessary to get an autonomous car to run correctly. What is required here is a real-time learning algorithm, not a pre-learnt set of rules based on which decisions can be taken. So, what is a real-time algorithm?

What I have described in the above examples are offline learning algorithms where the machine learns and stores a logic from some training data. But, real time learning algorithms need to learn on the fly, from real-time data. In an offline learning algorithm there is a training data which pre-programs the entity with a certain "learnt logic" and subsequently any other input data given is only used to generate an output based on the previously learnt logic. In a real-time learning algorithm the most relevant learnt logic needs to be adapted and applied to the real-time data. The reactions and responses due to the adapted and applied solution is taken as a learning to modify the logic present, in real-time and reused when the next relevant situation that occurs.

The requirements of such an algorithm is different from that of an offline algorithm. In a real-time algorithm world, data is streamed continuously. All data is not available at once to be sliced and diced and re-used and re-programmed according to the requirements. Information required has to be extracted from the data and computations done as the data flows through the different states of the algorithm. It further has to be fast enough to work with a throughput that is comparable to the rate of flow of data. In such algorithms having complex computations during the flow makes it slow. Having computations across a large set of data also reduces throughput.

It is also required to be able to search and locate relevant information rather than accurate or mathematical match for situations. In such a search and locate logic, adaptability of information becomes very important to apply to the given real-time parameters. The current learning algorithms have little or no adaptability present of the learnt logic. For example in the car driving example, let's say the algorithm had learnt the logic to project the trajectory of a second car travelling in the same direction as itself, based on its speed and wheel orientation, subsequently detect the position where its own trajectory will intersect with the second car's trajectory to take a decision whether to change langes. Now, if it encounters a situation (not learnt) where it finds a car is

crossing over from the on-coming traffic lane into its lane and is coming in a head-on collision course, it should be able to pull up the lane change algorithm as the most relevant logic, compute the required trajectories of both the second car and itself and adapt the decision taken to change lanes to prevent an accident. Take the output of such a decision and put it back into its own algorithm as a learning. This is a real-time learning and this is learning the skill to drive a car rather than the logic or rules present in data. It should be seen that skill accumulates over a period of time rather than staying put at the same level till the next training with more data.

**Rigid, digital data representation**

It is strange that we have terabytes and terabytes of data in each and every domain of expertise and yet we are not able to create a single intelligent being even a single domain that can handle all the nuances of that domain. With all the processing power present and terabytes of data present we are only able to train a very narrow intelligence even within a single domain. What is even stranger is, while there are terabytes of data, they encompass less information by multiple orders. Our representation of data and information is bloated by many orders of magnitude. Even in this paragraph to convey a single point I find I have a need to write multiple sentences. Even if we represented each character of a sentence with a single byte, we find that to convey a single piece of information i.e., to simply say “data representation is complex”, I need 29bytes of data.

But every potential particle in nature seems to embody infinite possibilities of data. It is time we started questioning how we represent data. We do not have a concept of what I call true “quantum data” where data can assume “any information that an observer chooses to give it” and until such time, the data remains indeterminate. We need to look at “a quantum particle” in nature with this perspective. While the scientists say that a quantum particle exists in all states unless observed, at which time it assumes a single state, what we need to ask is what does “all states” really mean? A different perspective of this should be that “a quantum particle” does not exist at all or “exists as indeterminate” until an observer observes it. The state where “nothing exists” is the state of shunya and can be seen as “data with infinite possible qualities”. Here, the association of qualities is to the observer rather than the data. Hence, the data itself can appear in different forms to different observers *[I have explained more related to this in Chapter 3 in the section on “Effective data representation for the problem at hand”]*.

In the computer world, the data that we create is tightly coupled to just one property, namely a binary representation of ‘1’ or ‘0’ which inherently has no meaning w.r.t the data we want to store. Hence to represent any arbitrary quality we need to combine multiple of these binary representations and associate a meaning to that representation. This representation is not a direct translation of what the quality is. It is a direct representation only of mathematical value. Hence behaviors of a quality that is not mathematical in nature needs to be mimicked to achieve anything in computers.

Moreover, data is represented as discrete rather than an continuous analog representation. When we talk about analog data, we cannot use individual data representation as we do in discrete data. For example, when we look at a sine wave, when we have a discrete representation of data, we would have taken samples of the sine wave at regular intervals, say every $15^{o}$, so sin(0 $^{o}$), sin(15 $^{o}$), sin(30 $^{o}$) and so on and stored it. Now, if we wanted to shift this to an analog data we cannot sample it. It becomes continuous data. So, how should the data be stored? We need to store it directly as a pattern, only then the continuity is maintained. So, we can very easily get sin(15 $^{o}$) as well as sin(15.00001 $^{o}$) as well as sin(15.19000 $^{o}$). But when we start representing data in this manner i.e., as an analog value, then we start having problems with data computations and logic, because what we need here is a pattern algorithm rather than a mathematical computation.

All the logic we write is oriented towards a digital representation of data. For example, the collection functions such as sort, search and so on are dependant on the fact that data points are available and are not patterns. The learning algorithms that we write are based on the fact that data points are streamed, hence we need to do clustering logic, regression logic and so on. In-fact, if we look at mathematics, we model functions to be dependant on variables which we assume are discrete in nature and computed only for specific values. We do not have the concept of analog logic, analog computation or anything related to analog unless they are technology created with materials found directly in nature. For example, electronics are basically analog until we force them to be discrete for easy computation. Thus, we have sacrificed continuity for rigid, discrete representation that has taken away a lot of functionality.

**Missing continuity in the algorithms**

We find that whether we are a child, youth or old, learning is something that does not seem to stop. Even if a person has

alzheimer's, we find that some adaptation and learning is always present. We can have forgotten some of the information learnt, but learning new things, however insignificant it seems, does not stop. As I have explained before, in our AI systems, given that we work with offline learning, once an algorithm has been formed with training data, the algorithm does not adapt itself for the data, to apply the algorithm learnt. More over given that our data representation is discrete, the logic is also discrete. Multiple discrete business logic tied together to create the illusion of continuity.

Let us take an example of an algorithm that we build, implementing a finite state machine. This is a very common problem that is implemented in every business application to track the state of the business object such as order, shipment, Part and so on. Sometimes we need to use multiple statuses against the same business object to track various functions.

For example: we have all seen an amazon order go from status of "order placed", "dispatched", "shipped" and "delivered". At the "shipped" status we can further track the shipment that changes the status from "carrier picked up at facility", "arrived at facility", "out for delivery" and so on. These are very simple finite state machines. Yet, it should be noted that the state of the "order" is contained with the information in the data associated with the order and other related business objects. It cannot be a secondary field stored on each business object that needs it to be explicitly updated. In the example of the order, the fact that the order is dispatched is truly a set of actions based on the warehouse operation. In the simplest, it consists of actions: "planning the shipment in which to ship", "picking the item from the warehouse shelf", "packing the item", "placing it in the bin that is allocated to the shipment", "loading the truck by the carrier based on manifest", "check the loaded shipments" and "truck leaving the facility". These are all coded as discrete logic and tied together by id relations that ties the data together.

Looking at this from a different perspective, if we were able to continuously track the movement of the "item" as it moved across the warehouse floor, we would have got the status without having a discrete field that represents the status of the order, which tends to introduce bugs into the system. The problem we all face when we write a software application is that "the data that is stored in the computers are devoid of the actual reality".

We are currently trying to fix the problem by introducing IoT sensors. But again these sensors are also discrete. We can introduce an RFID tag on each item and add RF readers across the warehouse to track where an item is present in the warehouse. But, we have still not got away from the fact that we have data that is discrete disparate data that we need to tie together. So relating the order and the item is a logic based process rather than a data-based process. Thus, all our logic is also very discrete in nature. We write code that tracks an item hoping to the good Lord that the person whoever is supposed to use the system to enter the details does a good job without entering it with errors. We have introduced many easy to use mechanisms such as barcode scanners and RF scanners that needs to be used, but yet errors creep in. We write code to take this item scan and update the shipment in which it is supposed to go, then we pick all of these and try to update the order with the status and so on we string together a list of business functions that are discrete in nature and again hope to that good Lord that the programmer has done a good job of handling all error conditions. No wonder, that this is the kind of code we are trying to get our AI to learn, obviously the AI also spits out discrete pieces that are strung together as a if-else clause.

The other version of the problem of not having continuity can be seen when we look at the example of autonomous car driving that I described in the previous section. As I have said, our processors are only capable of discrete processing. While we talk about it as a streaming, data is not real-time streamed, it is sampled at a very fast speed to simulate real-time streaming. So, if we added a sensor to know the current scene around the car, then we can only sample this data every 1s or every 0.5s or even every 1ms and process the picture captured by the camera. So, in a situation where cars are travelling at very high speeds the response time required to avert a collison where a sudden change occurs is very difficult, plainly because the data has not been sampled that fast. To do this, we need what we create in computer applications as event based processing. We need to be able to detect disruption in continuity rather than regularly monitor the scene by sampling. But, what we miss is continuity in our algorithms because of the discreteness of our data representation.

**Accurate computations as a necessity in our algorithms**

Taking the car driving as an example again, we find that when we drive we do not do any accurate speed, accurate angle for turning steering wheels, accurate computations of pressing accelerator etc

computations. We only know we want to increase speed or decrease speed and we appropriately press or release the accelerator. Same goes for the brakes. We do not compute the accurate distance the car will travel when we press the brake. We start braking gradually and based on the decrease in speed of the car keep pressing the brake continuously till it comes to a complete stop within the distance we want it to stop. Very rarely we know the mathematically accurate distance it has taken to stop the car. There are definitely those situations where we hit the brakes at full hoping and praying to the good Lord that the damned car will stop in time, but again we have no clue of the mathematical accurate distances, deceleration or any such numbers. Even here, we just know the rate at which the car is decreasing approximately and we may not be able to stop in time. Yet we can drive the car much much better than any computer program. Why is this so?

In reality, we rarely find the requirement for mathematical accuracy. Say we are drinking coffee, we just lift the cup and bring it to our mouth and tip it correctly for the correct flow of the fluid to be able to swallow it. In fact we can very easily control the flow rate of water or any fluid to be able to correctly keep swallowing in real-time as the fluid flows without actually know the accurate rate of flow of the fluid. The match of the rate of flow of fluid from the cup to the mouth to the throat is something we can do very easily. How is this, when in a computer even though we can easily compute accurately and match up speeds, we cannot use it to correctly do many simple tasks.

The point is that accuracy is not a necessity for knowledge and intelligence. In fact knowledge and intelligence is more driven by fuzziness and inaccuracy rather than accuracy. Learning algorithms does not need mathematical accuracy. That is only required for computations. What is required is patterns and pattern related algorithms and operations which we cannot easily do with computers. Pattern operations introduces as a part and parcel of it a randomness and fuzziness which cannot be got when we have an accurate computation machine. Thus, we find, when we need to have alerting programs that needs to alert when the temperature has gone beyond $35^{o}$, we need the accurate value of 35 before an alert is raised. $34.999999^{o}$ will also not raise an alert. We do try to mitigate this by adding Gaussian curves and other such solutions, but this is just band-aid. This is where inherent approximate comparisons of patterns and pattern logic makes a lot of difference.

Why does this make a difference in AI? When we look at the car driving example, say we computed the trajectories of two cars and determined that the collision is happening at a distance of 500m from the current point, should we change the position of the car or should we not change the position of the car? At what cut off distance should we start changing the position of the car for it to correctly respond? Should we spend precious compute cycles to compute the exact mathematical distances at which the trajectories will meet? Does it even make sense to compute it at that accuracy when we do not know with exact accuracy the length and width of my car or the second car? But, then say we did do an approximate computation, where does the approximation end? How much do we approximate?

**What do we need to create an AI that does not seem fake?**

What need to understand in all this is that we are trying to mimic what appears to us as the learning algorithm in order to mimic the intelligence of the universe and pass it off as AI and hence everything seems fake. To truly create an intelligence, we need to understand the internals of the working of the universe and find a way to implement that and allow the other related parameters to develop on its own, to get a truly intelligent experience.

We need a redefinition of what artificial intelligence is. A decision making system is not an intelligent system. A if-else rule based system is not an intelligent system. A system that can learn a logic for a given data and apply it to other similar data set is not an intelligent system. As I have said in the first chapter, all of these are just appearances or a result of the true underlying knowledge that is needed to create an intelligent system.

We need to create a system that can truly start with nothing but some seed information that dictates its own characteristic. Be able to put the system into a situation, gather knowledge about the situation, apply its characteristic on that situation to come up with a solution to handle the situation (problem or no problem does not matter), implement the solution, find the reaction, record it and keep collecting such reactions and as it keeps moving forward be increasingly efficient in the reaction to a situation. Subsequently, it needs to be able to pickup the correct relevant knowledge for a given situation and adapt the learnt solution for the varied parameters of the current situation.

In pursuit of this I started looking at the Sanskrit literature to find how does our reality work? This is what I have described

further in the book. The algorithm that has been described in these Sanskrit literature.

# Guiding principles of the algorithm

We all know and have seen that an app on the phone is not just the presentable UI that is shown on the phone. It has a whole lot of backend processes running on some cloud server somewhere, a database to store the state of the data running on some other cloud server and if it is a finance app, it potentially interacts with many other cloud-based applications running in many other cloud servers that belong to many other banks, payments gateways and so on. The app that is running on the phone is just a front facing "user interaction tool" that abstracts a network of applications that is doing the actual job in the back end. So, why do we assume that "we as beings" with more complex functions than even the simplest app on the phone will be purely this presentation layer of the body with all its organs? The first and foremost to understand and accept when we start asking ourselves "what is the algorithm of this universe?", is that, this universe is NOT JUST what we observe around us. Even the visible "sensory perception" themselves are not just what is presented. They are but a string of invisible observations that are not consciously acknowledged by our brains. While the eyes see light reflections, electrical signals are transmitted to the brain, but, we infer an image from this string of "invisible" observations. This, that is observable around us that is consciously recognized and acknowledged by us, is just similar to the presentation layer of that app on the phone. We need to recognize that we need to study and understand the internals of the working of reality around us for which this "sensory observation" acts as a presentation layer.

We can study the internals using the output from the "sensory observations" that we have. But, as I have said in my previous book "Surya Siddanta: The emergence of empirical reality", any "observation" can be only done *AFTER the fact*, not during or prior to the occurrence of any event. Hence if we want to study "How the internal functions of the universe is working to generate this appearance of observation", using information gathered from observations which is "after the fact", it is plainly not possible. The information is just not sufficient. We need to find a way to gather information from this universe around us "before the fact" or at least "during the fact" to study the universe. As I have said in the book "A research of Shiva: The Enigma", this cannot be done by "study using the manas". This is the major problem with the algorithms created for AI. We have tried to create an intelligent system using algorithms

formed with information which we have gathered using the “manas” which is just about the appearance or presentation layer and the information is “AFTER the fact”, rather than that which causes the manas to be formed, which is the actual internal algorithm driving the formation of the manas.

To understand and accept that we are emulating the wrong information, we just need to look at the various technologies that we have created. We will find that in not even one technology, that which is visible externally, is the same as the actual internals of the working of that technology. For example, if we look at the working of a battery, the observation says that if we connect the positive and negative terminals of a battery to a torch light, when we switch on the torchlight, the light should glow. The common observation is just the glowing of the light due to the act of switching on the torch. But the actual cause of the light glowing is neither the act of changing the position of the switch, nor the voltage developed between the two terminals of the battery. It is the flowing of the electrons from one terminal to the other when the two terminals are connected through the resistive material, that generates the heat and light. The resistive material reacts by generating heat and hence the light glows. If we wanted to emulate the behavior of a torchlight, we cannot do it by just simulating the voltage difference or the switch. We need to emulate the flow of electrons, which in-turn generates the voltage and then emulate the action of the switch that connects the two terminals to allow or not allow the flow of electrons. We need to emulate the high resistive behavior of the element such as tungsten to react by generating heat, only then have we correctly emulated the torchlight.

Another sample can be seen in driving of a car. While the car has a steering wheel, the clutch, brake and accelerator as the interaction points with the car, the car itself cannot be driven without the engine, the gears, the transmission to the wheels, the fuel that gets the engine to operate and so on, many hidden parts working integrally together. Trying to emulate a car by just creating the car chasis, the wheels, the steering wheels and the externally visible pieces, gives us only a model of the car, not a drivable car, which is our AI. A model of reality without the internal engine that drives that model.

Thus, if we want to emulate the human intelligence to create “artificial intelligence”, we need to emulate not the observation, namely our intelligence, but the internals of the working which generates intelligence as a final outcome. Intelligence is similar to

the voltage that develops between the terminals of a battery because of the differential in the collection of ions between the two terminals. Thus, intelligence should not be the primary artifact created. We have to write an algorithm that lets intelligence to develop. Learning is similar to the switch in the torchlight that allows the terminals to be connected to allow the electrons to flow. Thus even emulating learning is not going to get us a true AI that emulates human intelligence. To really emulate human intelligence, it becomes imperative for us to understand the internals of the working of reality around us.

In this and the next few chapters I explain how the algorithm of the universe works to give us this appearance of reality. Subsequently I show how this can be adapted as an algorithm to atleast start experimenting to create a truly AI system. Throughout the explanation of the algorithm, I have tried to bring out the differences in the way we have represented and created the algorithm vs how it is in the algorithm of the universe and why we need to reconsider and redesign even the system that is used to achieve AI. Before we start to understand the algorithm, we need to understand a few concepts that drive and form the guiding principles throughout the algorithm. I have explained this in this chapter.

## The starting point (The Brahman)

To set the context, before any other concepts. When we want to look at the internals of the working of reality, we need to understand that currently, it is in what we typically in the computer world call “the maintenance phase”, i.e., where the reality has started and is in the continuous phase of just continuing to exist. Yet, we need to realize that every new event that forms *starts at a state where nothing exists related to that event*. Further, what should be realized is also that the sequence followed in the formation of a new event within an already existing reality is the same sequence followed to create the very first event at the start of reality itself, when reality did not exist. This creation of the very first event at the very start can be thought of as a specialized case for our understanding. In actual when we see the internals of the working of reality, we find that that it works in a certain manner that just adapts to all scenarios, irrespective of our description or understanding.

So, the start is there “where the truth has not yet been mapped into knowledge”. If we are looking at the start point of a “new event” in an already existing reality, then the start is there “where the

truth has not yet been mapped into knowledge related to that event". For example, prior to the "industrial revolution", any knowledge related to "the wheel" did not exist, even though other related knowledge could have existed. Thus, for the wheel and all other events that occurred subsequently, the starting point was the "first conceptual thought that occurred of the wheel". To be understood and clarified here is the concept of the "first conceptual thought". As I re-read it, I find it is very easy for us to mistake the words, because in our minds we always associate "conceptual thought" to be that which is formed in "some human's brain". But, this need not really be the case. Many a time, if we follow through the chain of formation of a thought backward to find what triggered a thought, we will find that the root trigger could very well have been external to us, where an observation in the external world triggered the root thought in us. For example, the very famous example of Newton's discovery of gravity started with the "apple falling from a tree", not even sure if it is true. Or the other example is of Galvani's "animal electricity theory" was triggered because of seeing a frog's leg twitch to an experiment he was conducting. So, when I talk about the "conceptual thought", we can trace it back all the way back to the start point where a single insignificant change caused a cascading and accumulation effect to generate a concept.

To understand the starting point of reality itself, we have to see truth similar to that phase where we have "terabytes and terabytes of raw data in a NoSQL database" that is collected by the various applications to be used in data analysis, before any actual meaning or analysis is run on the data. But, there is a difference. While the terabytes of data that we have collected has a rigid format, rigid division and pre-defined meaning and a set of pre-defined qualities associated to it, that which we call "the truth" or "The Brahman" in the formation of reality around us has no such assumptions. Each and every "something", if such division exists at all, is only because of us, as an observer and if any such thing has a quality at all, it is also because of us as the observer, which has not yet been created. Hence, it can only be considered as that indeterminate, continuous whole.

Hence, to continue further in understanding, we need to form this nebulous, indistinct something, a point where no observer exists, no observed exists neither does anything that is an observation exists. Again why I say a point, it should not be mistaken for a geometrical point or a point in time. As Richard Bach has said in his book in illusions, it "is". This is not to say that what exists is

“nothing”. It is neither “void” nor “vacuum” as is usually described when “no matter” is present. There is “something” which we cannot know, *because when we start trying to know it, we form thoughts, then it just became reality in all its glory*. This state is called “Shunya”, where “*WE do not know what is, but something is*”.

Such is the starting point that we need to start at. Making no assumptions, making no observations, knowing nothing, applying no intelligence to it. The Nasadiya suktam says this about this state:

*na asad AsIn nO sad AsIt tadAnIṃ na AsId rajO nO vyOmA (a)parO yat |*

*kim AvarIvaH kuha kasya sharmann ambhaH kim AsId gahanaM gabhIram || 1 ||*

*na mRtyur AsId amRtaṃ na tarhi na rAtryA ahna AsIt prakaitaH |*

*AnId avAtaM svadhayA tad aikaM tasmAd dhAnyan na paraH kiM chanAsa || 2 ||*

*Translates to*

*At that time, neither the unreal nor the truth existed. Neither the space nor ether existed. Beyond what and within what was born? Where and under whose protection was the power? How this existed is an inexplicable mystery. Neither perishable existed, nor imperishable at that time, neither night nor day existed, nor it had appearance, breath untroubled inherent that one therefore that one none other like that beyond*

While here it is referred to as “that one”, Shankaracharya slokam says that you cannot even consider this as “*one, for where is the other*” and I tend to agree. The problem is when we try to describe something of such as this which is quality-less (nirguNa), in-coherent (nirvijnana), un-understandable (nirvikalpa), where even adding a description, brings it into existence, we cannot call it anything or describe it as anything. So, I will simply leave it at “The Brahman”. “The Brahman” is the starting point for anything.

## Effective data representation

To introduce the concepts, we need to start with the fundamental concepts. The first fundamental concept that we need to (re)consider is data representation. In-fact maybe we need to redefine what “data” means also. But, I will for now continue to call it data. It is

not “information”. It is not “knowledge”. It is not “conclusions or intelligence”. It is similar to what we have as “bits and bytes” that can be combined together to associate a meaning. Hence, I will call it data.

We need to acknowledge that computers with the current digital data representation of bits and bytes, were created with “mathematical computations” in mind. We are trying to currently write a system that mimics “intelligence and logic” with that same bit and byte representation. We have never asked ourselves “can the same system be used for both a mathematical computation application and intelligence and logic application”? So, what is the difference in data representation that is needed between the two? We need to ask ourselves, “can the same data representation and computation processing capability that is present in our computers be effective for all types of use cases”? To understand the effectiveness of a data representation, we need to understand the differences in the use cases between the two. Comparing the use cases we find a large number of variations in requirements for data between the two as I have explained.

In a mathematical computation based system, we know the algorithm already and many of the computations are already wired into the processor of the computer, such as increment, decrement, add and so on. In an intelligent and logic learning based system, we do not necessarily know the algorithm upfront. It has to be learnt. The data that we have to learn from is just “*ONE set of possibilities*” of many other sets of possibilities. We are searching for intelligence and logic within the data that can be applied to other similar or dissimilar data sets. In such a system, more than the “*accuracy*” of the value represented by the data, the “*intent and context of the data*” is more important.

Mathematical computations are based on *operations on discrete single valued datum* and relations are established by *well-defined mathematical functions* that can be applied and computed by replacing *known variables* with different discrete values. So, if I wanted to write a linear regression function, it has a equation of the form Y = a + bX, where ‘a’ is the bias, ‘b’ is the weight, ‘X’ is a dependant variable and ‘Y’ is the computed variable. Here, discrete values of ‘X’ will produce related discrete values for ‘Y’. Intelligent and logic based systems require continuous and highly related data with *unknown variables*. The continuity also represents strength of the intent and relations depict the covariance of parameters in relation to each other. For example, even for an AI to drive a car on

a empty stretch of road, the data representing the road on which the car is being driven needs to have continuity to ensure the car can follow the contours of the road steadily without missing curves. The continuity has to be broken only when an obstacle appears on the road. In which case, the continuity of the obstacle needs to be present for the extent of the obstacle after which it passes to be replaced with the original road's continuity. We definitely can see the road as discrete, very closely sampled data points which we tie together to form the continuity, but obviously we could have missed a curve just at the non-sampled interval which will lead to unnecessary conclusions or at the minimum lead to lane drifting. There are many ways to solve the problem, but, they are all just work arounds to get around the continuity problem of the data representation.

Trying to represent continuous and related data with discrete values occupy lot of space and require *intensive logic to maintain the simplest relation* such as subsequent or sequential values. Again, we are in a domain where data is not always mathematical in nature. Hence, sequential values need not necessarily imply subsequent value. In computers when we define data, it contains only one set of meaning that can be correctly associated to it. So, if we declared a variable called a byte and associated it with the hex value of 0x41, then, it only represents the number 0x41 or 65. If there is another program that wants to interpret this 0x41 as an ascii character, it can be interpreted as the character 'A'. But that is about it. It remains at 0x41 and interpreted differently and can only take on say two discrete meanings. Now, let us add another byte and associate it with the hex value of 0x42, this again only represents the number 0x42 or 66 which can be interpreted as 66 or 'B'. Intent-wise, these are sequential only when looked at in terms of the series "integer values", this is the next value to 65 or when looked at in terms of the series "ascii characters", where 'B' is the next character to 'A'. When the intent is of a meaningful word, typically 'A' and 'B' don't occur sequentially in a single word. Thus, sequential and subsequent have different meanings.

To represent and be able to understand this relation, either, we need to have a hardcoded logic in the system that force the two bytes allocated to be co-located next to each other, by changing the byte alignment of the declaring class of bytes or by using an ordered array or any such elaborate mechanism. Obviously, this ties us to the compiler, the language and all the other parameters that allow us to write such a code. In any case, it boils down to the logic coded by

the application, rather than the data representation automatically providing us with inherent capability for relations. The discrete-ness in our data representation is not just a feature of the way the software is written, but is a feature that boils up to the software layer, all the way from the hardware representation of data and impacts the processor that is processing the logic. Can such a representation be effective for intelligent and logic based systems, where even the simplest logic of having a knowledge of series of sequential data, needs a lot of code to be written to establish it?

Let us compare this to the "data representation" present in the underlying truth or what is called "**The Brahman**" in the Sanskrit literature, that forms the inner workings of reality. Currently, in any science, we do not have a word nor do we have a description for the representation that exists in the underlying truth. Hence I am using the word "data". We also do not have word in English for describing what is called "The Brahman" in Sanskrit. "Brahma" translates to "create or creator", thus Brahman is "that in which can be created from". So, as I have said before, I will continue to refer to it as "The Brahman".

As I have explained in the previous section, "The Brahman" is that continuous whole which is in a state of "an expanse of Shunya or non-existent nebulous something". In this continuous whole, an "*insignificant bit*" of this Brahman can take on any property with a value. Hence from that continuous Shunya, a bit of it can take on the states of activated or inactivated which becomes data. Many such insignificant continuous bits activated for a specific property with different values come together to form a concentration, which is considered as "continuous data" or "series of data" of that specific property.

To understand this, if, we look at "The Brahman" as a huge painting canvas which is ready to be painted. In this huge canvas paint of many different colors can be painted in insignificant bits, which when "viewed" together becomes a whole painting. In this canvas, we find that, some paint bits coming together, forming the main focused picture such as a portrait of a person, while some other paint bits form the background in which the focused portrait is present. These background paint bits now can be split into data that are active from the main focused picture and inactive from the background. Same color bits can come together to form a concentration of the same color or different colors could have melded or diffused into each other. There can be pieces within this painted bits that have neither been "activated" or "inactivated" and

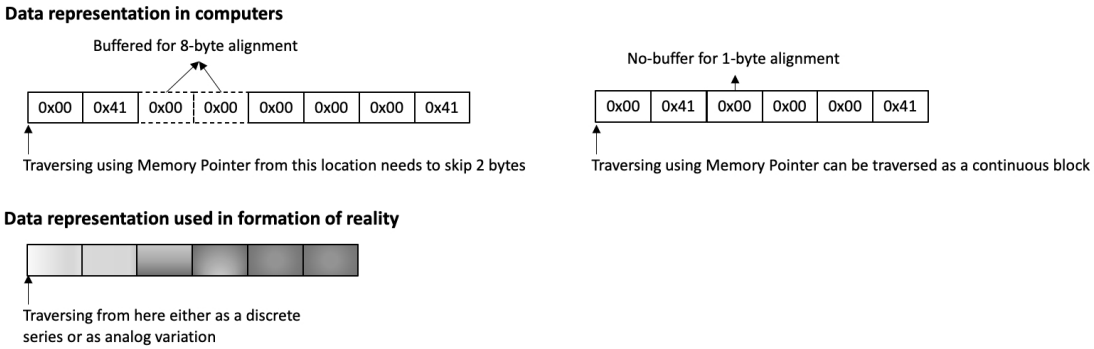
still have the background canvas visible as is. Similar to this is the data representation that is present in “The Brahman”.

Some major differences in the way we represent data in computers to the way data is represented here, is: Here, we start with a continuous expanse that is then broken down to bits that represent data as opposed to having discrete data representations that need to be combined together to get the continuous expanse. Further, inherently, the bits of the continuous expanse can be either active or inactive. This gives us a data representation that has inherently two-states, independent of the value they take on. Along with this, data is a data only if it is associated with a property type and a property value. But, what is the most important piece to this is the “observer’s role in creating the active or inactive data and associating a property type and value to it”. As I have explained later on, a data of a specific treble (property type, property’s value, state) is present due to the presence of an observer associating the treble. The same continuous bit can take on different trebles based on the observer. Thus each data, inherently is represented by the quadruple (observer, type, value, state).

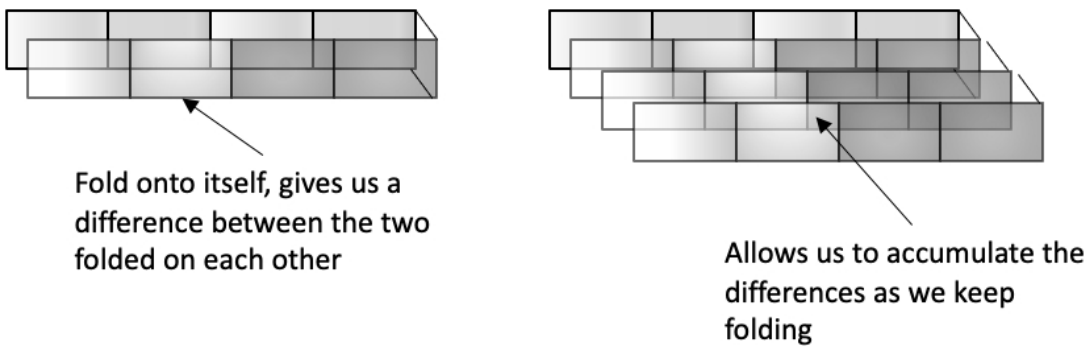
What are the advantages of such a “data” representation? We need to realise that while it is broken down to continuous bits because of the observer’s ability to create the illusion of discrete-ness, the original continuity present in the Brahman is not lost. So, a series of “discrete bits of data” with subsequent sequential property values can either be seen as just that, i.e., is a series of discrete values or can be seen as an analog stretch from start to the end of the series with a gap separating this analog stretch from the next analog stretch. Thus, the same observer can interpret the same data in different modes based on the need. Either by splitting into series of bits or by combining the series into continuous bits based on what is needed for interpretation.

To look at a comparison to understand this: In computers, typically data structures are aligned on 8-byte boundaries for faster access and optimization. So, a data structure that is declared with a short (2 bytes) and integer (4 bytes) as its member variables, has a size of 8 bytes with extra two bytes added to the short variable. If an pointer to the address of the storage location is traversed from the first byte to the last byte, it will not continuously have valid values that can be used by the program. But, the data structure can be forced to have a 1-byte alignment when compiled. This forces the structure to have a size of 6-bytes. Here, the same memory location that stores the data structure can be interpreted as a structure with

"short" and "integer" values or can be interpreted as a series of characters of 6 bytes. Similar to what is shown below.




This is exactly what the "data representation" in reality allows us to do. It also takes away the the discrete-ness of the single bit/byte that we are forced to work with in computers. This is shown in the above diagram. We can see it as a continuous analog variation of values for a specific property type, which I have represented as a gradient of colors. Since, this representation has embedded in it the ***contiguous blocks of data*** that is representative of series data, it allows us to fold a data series on itself. Given that data is just a quadraple of (observer, type, value, state), the observer observing the folded data through both the folds gives us the various deltas across various values in the series, that can be represented as the same quadraple for the observer, as shown in the below diagram. If we keep moving the fold, we can get the deltas between all values in the series. It also allows us to accumulate it easily to give us integrals represented in a data format which is the same as its own data representation i.e., a quadraple (observer, type, accumulated value, state).


Fold onto itself, gives us a difference between the two folded on each other
Allows us to accumulate the differences as we keep folding

This type of representation has embedded in it a perspective which allows the same data to be spun into different perspectives by different observers, giving an opportunity to accumulate across

observers the same continuous bits to gain different perspectives. While, I have listed the main advantages here, there are many other advantages of such a data representation. I have explained these as I explain the various states the data can take and as I explain the algorithm.

## Inherent states of data

As I have explained above, the representation of data is such that it can inherently be in four states, the unobserved or just the continuous Brahman, the indeterminate state, the dormant or inactive state and the active state. In quantum computers we talk about qubits in three states. While the normal state of the qubit can be looked on to be similar to the indeterminate (called coherent superposition), the qubit can fall into just a determinate "1" or a "0" binary value when used, as opposed to the infinite type/value representation that we talk about in the inner workings of reality. So, there is no true indeterminancy in qubits as compared to the "true indeterminate state" talked about in the data representation in the inner workings of reality. Thus there are only two states to the qubit, the coherent superposition state (where it can take on "1" or "0") and a determinate state that can have a value of "1" or "0".

### The indeterminate state before data (OM)

While I have said that "**The Brahman**" is a continuous expanse of a whole, when I look at the Sanskrit literature, I find that they do not exactly talk about the continuous whole called "The Brahman" as being the source of all this around us. The Sanskrit literature actually talk of the "OM" as being the source of all this around us. The OM is consistently used everywhere in the Sanskrit literature. It is time for us to try and find out what "OM" possibly could mean. My own interpretation of "OM" went all the way from thinking it to be a vibration to a 7.3Hz frequency wave to accepting it to be some (unknown) concept forming this underlying truth that in-turn forms this environment within which reality is formed. But, none of it really fit into the interpretation of all meanings in which it has been used in the Sanskrit literature. I think, the best description of OM is given in the Mandukya Upanishad which says:

*OM ity etad akSharam idam sarvam, tasyopavyAkhyAnam*

*bhUtam bhavad bhavishyad iti sarvam OMkAra eva*

*yaccAnyat trikAlAtItaM tad apy OMkAra eva*

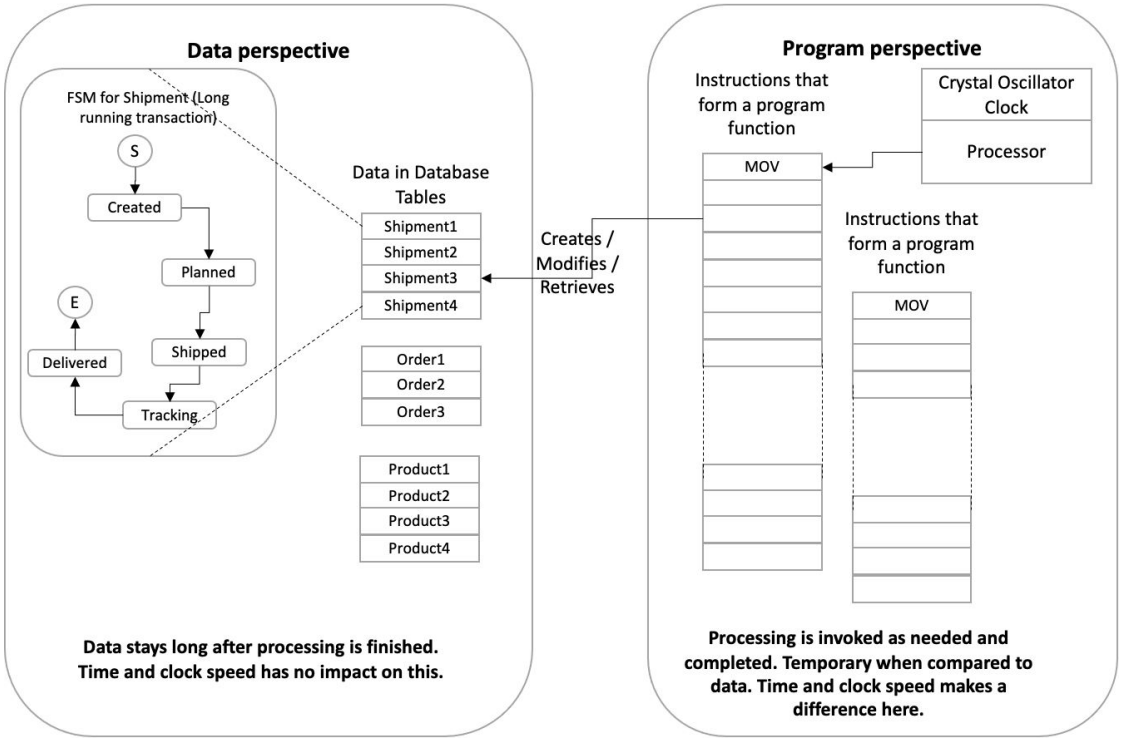
*Translates to*

*OM order of the imperishable here, is all this, from that which is obscure, that which has become, is becoming and will become here, is also the work of OM, that, any other (work) in the network of three series of changes, even that is the work of OM.*

For a long time I have wondered about this verse. How can OM be source of everything? If we thought "OM" as "an atom", then it comes near to encompassing everything that is what we call matter. But, "an atom" is very tied to the matter form of reality and does not encompass everything of reality which needs to be inclusive of both the observed and the unobserved (observed and unobserved by us). Hence, I tried replacing it with "a frequency of wave" and given Schuumer resonance is supposed to be at 7.3Hz, there was a time I thought it was this. But then, that again, does not encompass everything in reality. It only encompasses the observable reality that is formed of light elements. We cannot be sure that reality is composed only of light elements. Then obviously, there is the question of "the obscure", "the becoming" and "the become", how does any of these tie into "time" as we understand it. As I have indicated before, according to the Sanskrit literature, time does not exist. It is just a "series of changes" which is "kAla". We see that the above verse says, time (kAla) is also just OM which is very difficult to envision especially if we have not pondered over the question: "What is space and time and do they have any existence at all in the underlying truth?"

To understand this verse and OM, I started looking at the perspectives of a program running in the computers. A program running inside a computer has no concept of "time" at all. Time is only a concept that "*we, as the person who runs the program*" have, to measure the performance of the program. The only concept that is really of any use in the program is the concept of the number of clock ticks that has elapsed since some start point. We can very well see this when we add code to profile the performance of our code accurately. We measure clock ticks which is the highest resolution we can get compared to nanoseconds and milliseconds. A standard duration for "clock tick" exists only because of the processor's processing capability or speed computed by the clock present in it. It is possible that, rather than looking at the program from the number of clock ticks taken to process (which is very temporary, once the instructions are run, it comes back to run only when data requires to be processed again), we look at it from the perspective of the data that is being processed. We find that with this, we have a newer

perspective of what the program is doing. When we look at it from the perspective of various data that the program is modifying constantly, we find that even the clock ticks of the processor makes no sense. The only thing that makes sense is the changes occurring to the data as it goes from one state to the next. We typically call this as "long running transaction". So, we can have a Shipment data going from the state of created to planned to shipped to tracking to delivered to completed, as shown below:




So, what does this all have to do with understanding OM? We can look at reality around us as the flow of events sequentially from one event to the next or two parallel events occurring at the same time with many or one or no participants in each event. This perspective is equivalent to the program perspective shown above. In this case, we are looking at reality as various "*this instant processing*" which we call events that are happening. In this perspective, space and time takes on prominence, because, we are looking at events that need to occur in space and time, events basically being simply some logic executing. In this perspective, data is external to the logic, only included as a participant in the event with the event changing the data.

***Or***, we can change the perspective and look at reality with the "*data*" as the primary view and track its changes, with the event viewed as some logic that needs to run to change the state of data.

When we look at it from this perspective, “space and time” fade away and only changes of multiple “bits of activated data” come into perspective with space and time appearing only when events need to execute to modify the data.

In this perspective, the meaning and importance of OM starts emerging. We start with “The Brahman” first where even the concept of data is not present. In this there occurs “those bits” that can be activated or inactivated and has the potential to take on multiple properties and values. If we read the above verse of Mandukya Upanishad, with the context of Shunya and this perspective of inactive and active data, then, OM is equivalent to “***that bit from the continuous whole, which can be observed to take on any property type and a value and has the potential to become an activated or inactivated data***” i.e., a smallest piece of that continuous truth that can adopt any value. With this definition, the verse starts making sense, because, in this case everything is actually that OM in different forms. Thus, OM is “indeterminate bit” from the continuous whole that has the potential to become data. And how it becomes data is said in the Gayatri Mantra:

*OM bhur bhuvah svah tat sa vithur varenyam*

*bhargo devasya dhimahi dhiyo yonah prachodayat*

*Translates to*

*OM knowing to choose that one light energy to become, thence becoming self-aware with intellect, this being the origin of impulse*

The “OM bhur bhuvaha svah” is called the vyahritis and is very important, which translates by itself to “OM” + “to become” + “becoming” + “self”. Thus, this by itself has the meaning: *OM taking on a “become”, becomes the self*. This indicates that OM is “indeterminate” and choosing to take on some value to “become the self”, becomes the self.

So, to recap, *OM is that underlying “basic whatever” that can take on infinite possibilities of property and value, to become the data*. I don’t believe there is any word in English or any language that we know of that has this meaning. A quantum particle also represents only so much compared to this, because for a quantum particle to exists, the material realm of matter needs to be present, and the particle is adopting a set of property values related to matter. OM is something that is much much much before a quantum particle in the series of inner workings that form this reality.

**The active state of data (devA or Energy)**

As I have explained before, the continuous expanse can have "bits" activated to form active or inactive data. The first question that arises is: "how does a bit get activated"? The second immediate question is: "what *is* a bit of that continuous, is there a way to qualify it"? This is where it gets very interesting. The answer to these questions is such that, it completely decouples the data representation from what "The underlying Brahman" actually is.

To understand this, we need to search for the meaning of "prANa". Most Sanskrit literature I have read, talks about prANa, with reference to source of life. For example the Prashna Upanishad says this:

*Atmana eSha prANo jAyate*

*Translates to*

*Due to seeking for the self, prANa is produced*

Many other places talk about prANa in a similar manner. Hence it is understandable that prANa is translated literally to "life" or "breath" and so on. But, if we read the Surya Siddanta, we find that it talks about prANa, not as life, but as "a form of change". It says:

*prANAdiH kathito mUrtastruTayAdyomUrtasaMjnakaH*

*ShaDibhaH prANairvinADI syAt tatShaShTayA nADikA smRtA*
*||11||*

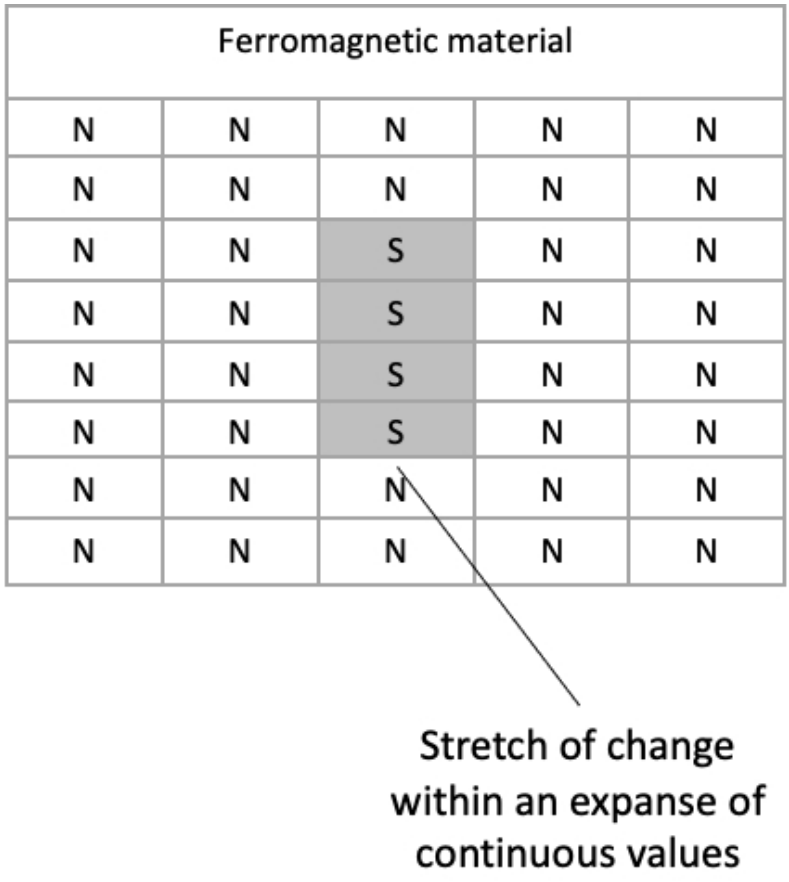
*Translates to*

*Related through prANa is form of change here, a harmonized primitive form, a sixth part of the thread of prANa is vinADI, sixty of that is nADi, that is recollectable.*

Here, it talks about prANa being a "harmonized thread of changes", which is the "continuous bit of data that we have". Thus, we find that in a continuous expanse of the Brahman, if a "stretch is detected that is different from the continuous expanse", then that "stretch of change" becomes "an activated bit" or "data". Change of what is pretty irrelevant here, because, the observer could have observed a change in any property, neither the property name or the absolute value makes a difference in making it a data. They are just caviars associated in the quadruple. The stretch of change within a continuous expanse of no change, is all that matters to make it a data. Thus, we find that because of this characteristic of activating data, the same continuous bit, based on the "observer's capability to observe" automatically takes on different property types and values

to activate as data, thus de-coupling the definition of data from the actual properties and values of The Brahman.

So, we could think of “The Brahman” as a “continuous expanse of magnetic material like a ferromagnetic material”. If we had an observer that could see the orientation of the magnetic fields to be all in “N” direction, but has a “small stretch where the orientation has changed to a “S” direction”, then that “small stretch becomes activated data”. As shown below:


Ferromagnetic material
N N N N N
N N N N N
N N S N N
N N S N N
N N S N N
N N S N N
N N N N N
N N N N N
Stretch of change within an expanse of continuous values

To be noted here, I have depicted here a contiguous expanse of same orientation namely “N” with a stretch of “S” orientation. This need not be the case. We can have the expanse of “unordered orientations”, a random orientation of “N” and “S”, with a stretch of just “S” or a stretch of some pattern of “N” and “S” and so on. Many different ways in which the change is effected. Again, it should be noted that I took this example assuming that the observer is one that reads magnetic orientation. I could add along with a magnetic orientation reader, an orientation of spin of the electrons in the material. In which case, the same stretch can be activated with a different meaning.

The next question we were asking is “what is a continuous bit?” If we look at it from the perspective of “change being the activation mechanism of data”, then it follows that a bit becomes that stretch with some pattern in changes till a change cannot be observed any more. In the above, the “bit” is with 4 blocks in it. With this as the definition of a continuous bit, we find that the extent of each data need not be the same across activated bits and depends on the observer’s ability to recognize changes. This is much different from out data representation. We have a blocks of pre-defined standard size that used to depict data, a bit, a byte (8-bits), a word (2 bytes) and so on. Based on whether it is a 32-bit or a 64-bit, the number of bytes that can be processed together vary, but again tightly tied to the type of processor.

To go further, the activated data is also looked at with some standardization for the sake of standardization of description and applying a common algorithm. But, as I have explained in the book “Surya Siddanta: Emergence of empirical reality”, the standardization is purely an abstraction and any representative values is self-contained within a data itself. This is measurement system the Surya Siddanta has described:

*1 vinADi = 1/6 of prANa*

*1 nADi = 60 vinADi = 10 prANa*

*1 orbit = 60 nADi = 60 * 60 = 3600 viNADi = 60 * 10 = 600 prANa*

*1 mAsa = 300 viNADi*

*1 varSham = 12 mAsa = 3600 viNADi = 1 orbit*

*1 thought = 7 orbits*

We see that each of this definition is dependant on prANa which is a recognized change extent, which can vary from one observer to another, from one property to another and many other such parameters. That recognized change extent is only given a structure, so that we can easily compare change extents with each other, rather than it being a rigid size of extent and change amounts as we have.

To note here: As I have explained before, once we have “data” from a bit of OM in that continuous “Brahman”, that data can be viewed as multiple discrete bits with property and value or can be seen as a continuous division. The advantage of seeing it as a continuous division is that it can be folded over and we can get easily the deltas. Taking this with the fact that data is actually only

changes, when we fold over and look at the delta of the changes, we are not looking at the deltas, but instead we are looking at differentials. This is used in the foundation algorithm, I have explained later.

Relation of activated data to energy: As I have explained in my book "Surya Siddanta: Emergence of empirical reality", energy definition in modern science is not a complete definition which is in and of itself. It is circular definition with work, where energy is defined as the capacity to do work and work is defined as the transfer of energy to move an object. Such definitions cannot go far when trying to understand the inner workings of reality. We need a more meaningful definition. The Surya Siddanta, gives the definition of energy in this verse:

*ityuktvAntardadhe devaH samAdishyAshamAtmanaH*

*sa pumAn mayamAhedaM praNAtaM prAnjalisthitam||7||*

*Translates to*

*Thus connected, the energies are held in between distributed portions of selves, that spirit consisting of these is continuous change evenly established*

From the above, it says when we look at the change across the selves as diffusion of various concentrations of continuous bits across a stretch in the Brahman, then that diffusion contains energy. So, if we looked from the perspective of a instant frame in that stretch of concentration, energy is whether it has the potential to vary in the next instant frame of that stretch. This is the very basic definition. When the variation takes on different patterns as the frames are traversed, different types of energies are formed.

According to what we were explaining above, a stretch of change across a stretch of continuous bit is what is "activated data". Thus, "activated data" is also "energy or devA". In our reality, we recognize the pattern that is related to thermal energy. The potential for a set of changes to start oscillating and take on the pattern of thermal energy is what is called "Agni" starts off the formation of reality.

**The dormant state of data (tamas)**

While we can talk about qubits as having three states of data, as I have explained, "0" is just another active value like "1". A variance in data value is not the same as the "state of data", that is explained in the Sanskrit literature. These are just considered as "values" that

can be taken on by the “data” when activated. Thus, a bit has only one state that is an active state with representation for two values “1” and “0” and a qubit has two states i.e., the uncertain state or coherent superposition and the active state with representation for two values “1” or “0”. In-fact, a bit also can be considered to have an “uninitiated state”, where the data can be either “1” or “0”. For example, when we use “malloc” in “C” to allocate a block of bytes, all the allocated bytes can have any value based on what the bits of the bytes allocated have stored previously, as opposed to a “calloc” that forces all values to be “0”. Thus, we can look at qubit and bit to have two states, an uninitialized state and an initialized state. In the initialized state they can have a “1” or “0” value.

As opposed to that, when we talk about “data” as represented in reality, it truly has three states. The first two states are: one the indeterminate state (OM) and two, the activated data state (devA) as I have explained. The third state is the dormant or inactive state. It is typically referred to as “tamas” or as “asura” in Surya Siddanta. “Tamas” is generally translated as “darkness” in Sanskrit. But, I think, in the context that we are reading these Sanskrit literature in, it represents “inert”, “inertia” or “dormant” based on the context it is used in. The Nasadiya Suktam of Rigveda tells us:

*tama AsIt tamasA gULham agrE aprakEtaṃ salilaM sarvam A idam |*

*tuchchy EnA abhva apihitaM yadA AsIt tapas as tan mahinA jAyat aikam || 3 ||*

*Translates to*

*Unbound, inert, foremost covered by dormant, all is indistinct, unsteady is this, minute in that enormous enveloped, if unbound heat exists, there in that mighty is produced that one foremost agitation.*

This tells us that there exists a state before the agitation (i.e., change) comes to activate the data and that state is “tamas”. So, what is this state? The Surya Siddanta explains this w.r.t the varShaM. It says that varShaM has an “active and an inactive version associated with it”.

*surAsurANAmanyonyamahorrAtraM viparyayAt*

*tat ShaShTiH ShaDguNA divyaM varShamAsurameva ca ||14||*

*Translates to*

*Active - inactive, inverse of one another of the great orbit there, that sixty's moving six qualities varSham is also inactive*

The Nasadiya Suktam and BHPS have talked about it as "tamas or dormant", the Surya Siddanta talks about this state as "asura or inactive". The Surya Siddanta further tells us, that the active and inactive are "inverse" of one another. "Inverse" has many meanings based on the context. When it is used in the context of mirror, "inverse" means a reflection of the image, when used in the context of functions in mathematics, it reverses the function's output and so on, based on the context, we get different meanings.

So, the question is when we are talking about active data, what does inverse mean? My way of understanding this is by looking at any electrical device we have. For example, a microwave can be in different states: "switch off and not plugged in", "plugged in but not in use, where it just displays the clock", "plugged in and in use". The switched off state, where "a microwave exists", but is not operational, it is dormant or inactive, this is the state of "tamas" or "asura" for the microwave.

So, the question to ask is what makes a data active or inactive. The obvious answer should be "the observer". If an observer can "observe" the data to give it a property type and value, it should be active and if it cannot "observe" a bit, then it should be "inactive for the observer". Again, note that "a bit" can be inactive for one observer, but active for another. So, the question is "how does an observer observe?" Knowing that should tell us how to identify the active and dormant data.

When we look at the old magnetic hard-disks, they used to have read/write heads that would seek a position and move across the disk to read or write the data. The seek is random, while the subsequent read is sequential. In this mode of operation, we could consider the head that moved to read as the observer of the data in the disk. A hard disk as we all know and might have encountered, can have bad sectors, caused for various reasons. But, the effect of the "bad sector" is that it cannot be read or written into by the read/write heads. Such "bad sectors" can be seen as inactive data. For example, one reason for a bad sector can be the "error correction code" has been corrupted, hence the sector cannot be read. It is possible that it has been infected by the malware which has modified the "Error correction code", in which case, the malware will be able to read the sector but not a normal program. Two observers, to one the data is inactive, while to another the data can be active.

This is in a magnetic disk with read/write heads as the observer.

So, how does this work when the observer, the observed and the data are the same, namely, “that bit of the continuous Brahman that is active or inactive”? The Surya Siddanta describes the observing of data as:

*toShitastapasA tena prItastasmai varArthine*

*grahANaM caritaM prAdAnmayAya savitA svayam*

*Translates to*

*Undisturbed concentration in a manner that grants complacency to that which acquires that which consists of giving motion by itself to the self.*

If we look at the above statement it is a little tough to get the meaning of it. But, we just need to compare it to some occurrence in nature to understand what is being said here. We find that nature works on certain rules that we cannot prevent very easily. Sometimes, we need huge amounts of energy to prevent certain natural rules from playing out. For example, the natural tendency of a river water is to flow in a certain course. To change this course or build a dam across the river, the amount of energy required is huge. Many a time, we find that when it rains and the water levels rise, the dams break or the original course of the river is restored. This can very easily be seen played out in Bangalore. Many low-lying lake beds are filled with mud and huge buildings built over it. So, every monsoon, these apartments and houses get flooded, because these are the catchment areas for rain water.

This is where this above statement acts as a rule using which nature acts. We can reason and describe the reason as gravity or terrain contours or any other geographical explanation we have. But, the general rule that is followed is that everything in nature acts to reduce the energy used to the least amount. Obviously, I have to say, excluding humans, who seem to always try to use the greatest energy to achieve something and cannot live in tune with nature. But, the question then arises, how does “the river” know the “least energy is used in a certain course vs another course”?

To understand this, we need to look at motion different from what we usually see it as. We tend to see motion as primarily a movement in space with time as a variable that effects the movement. But, motion in time itself is not considered. When we look at time as a series of changes rather than time the way we see it, then any change to the current parameters of the observable is

motion, not just space displacement. If at every frame of the river, if the river found the next minimum recognizable change to move to, then, given that energy is just the potential to change, the total energy used by the river motion will automatically be the least energy used. According to the above statement, it says, in an undisturbed concentration, we always find that a “self” follows that which gives itself motion, which is what the river is doing.

So, what has this got to do with inactive data. As we saw, when we have an observable, observing itself and computing and moving to the next change so that it can move, then that change which it did not choose, becomes “inactive data” for that observable. The other way to look at this is from the perspective of a pattern. A pattern cannot be formed from just one continuous bit. Multiple continuous bits are required to form a pattern or oscillation for the specific thermal energy pattern. Between multiple continuous bits, gaps need to be present. Gaps are unrecognized bits of changes by the observable. These gaps are inactive data w.r.t that observable. Thus, the same quadraple (observer, type, value, state) where state is inactive, represents even an inactive data.

## Data as primary driver of algorithm

The next fundamental concept to understand is “*data-based algorithm*”. A major difference between the inner workings of reality and our AI systems, is that the inner workings of reality uses a data-based algorithm where as our AI uses a logic-based algorithm. “Data-driven” is a concept that has been very recent term that got introduced in the computer world as a part of data analytics. This is not the same as the data-based algorithm that I am talking about here. Data-driven approaches drive decisions taken by companies based on intelligence gathered from data applicable to the business of the company. But the algorithms themselves are logic-based algorithms.

To understand the difference between data-based algorithm and logic-based algorithm, we need to understand “what is a logic-based algorithm?” The primary focus of all our computer applications is the “logic”. To get our computers to do anything we need to think of a logic or sequence of instructions that needs to be executed in terms of the language used by the computer processor and code it, to execute a program. It can be a computation of a mathematical function(s) or any complex logic that can be coded using the

language exposed by the processor. The data comes later and is modified and stored based on the logic coded.

In such a system, the logic comes first, then the data. The logic coded drives the state of the data. This is not purely because of how the software is written. The hardware of the computer is also primarily logic-based i.e., the processor is the main focus. Hence the software has evolved to being a logic-based algorithm. The processing provided by the processor works on data that needs to be loaded into their registers. It is also seen that the instruction set that is a language used to code the logic executed by the processor is mainly mathematical computation related instructions. Typical instruction sets consist of arithmetic functions such add, increment, decrement, multiply etc., binary arithmetic functions such as shift left, shift right etc., data transfer functions such as load, store, set etc., logical functions such as AND, OR, XOR etc., control transfer functions such as jump, goto which allow us to implement loops and if-else clauses, string related functions, I/O related functions and so on which are purely peripheral access control related instructions.

As we can see, the instructions related to implementing a logic are just binary in nature such as AND, OR, XOR that work purely on bytes. The logic control instructions are purely helpful for implementing if-else styles of programming. The data devices are considered as peripheral devices that are added-on externally based on the storage required. This can be accessed via the data transfer instruction set. We can write computation logic without storage devices also.

Given that the evolution to AI algorithms is from such a mathematically oriented computation device, and from the experience of writing logic-based applications, our AI engines is also are based on logic first rather than data. We build learning logic that use sample data to learn and build AI engines that learn “the logic that potentially modified one data into another data”. This learnt logic is then applied to other similar data. Since computer processors are purely mathematical in nature, we end up creating AI engines that learn a mathematical function, that can act as an activation function, or compute weights or co-efficients that again are plugged into other mathematical functions to compute new mathematical values. Thus, the nature of AI engines are typically numerical with a conditional logic that selects the correct mathematical function to apply in the correct order. This is what is called a logic-based algorithm. The logic comes before the data. The priority of the data is secondary.

The major problem with such AI algorithms is that not all data can be fit into a single regular mathematical function, hence the need for the conditional clauses represented by the neural nodes activating or not. Such a logic is rigid and can get hugely complex and convoluted based on the number of distinct data sets that needs to be handled. As the distinct data sets grow, the required conditional clauses also grow humongously wieldy. There are many types of algorithms to handle this conditional style of operation. The first is the most obvious, which is to write the if-else clause directly into the code. The problem with this is that it cannot scale very well, hence, as the number of conditions grow, the code can get very convoluted and complex to maintain. The second is to code a "map" with a data related key, which maps to the logic to execute. Here the key of the map can be computed from the data and the map looked up to get the logic to be executed. Such an implementation can scale and can be applied if the conditions based on which the key is computed does not become complex. The problem with such implementations is that prioritization of the logic cannot be easily done. Typical if-else clauses allow data to pass through the next and next conditional clauses which is lost if implemented using a map. The AI engines are implemented with both these styles of programming, the machine code being learnt using a learning logic. A typical conditional style programming becomes really complex, especially when the depth of the if-else starts increasing. Thus, when an AI neural network starts having higher number of layers, the complexity of the network increases multi-fold.

To add to this, our data representation is such that, data processing can only be done sequentially. While threads can be used to do parallel processing, it is not inbuilt into the data representation and can be done only for mutually exclusive data processing. For example, to compute integral of differentials across all the values in the stream, we need to individually compute the differential for each and every discrete data point gathered sequentially and then integrate. While data analytics use data-driven models to search out knowledge and intelligence that can be used to target advertisements, product marketing and other such manipulative applications, they are largely limited to the realm where user behavior data can be collected and analyzed. Yet, they also depend on mathematical logic to compute user behavior. They detect trends using linear regressions or statistics or graphs.

Typically, once learning is complete, the data used to learn is not retained. While data-analytics can reuse data, their reuse is only to

infer other extra knowledge or intelligence. Thus any data discarded as outlier, that comes back into focus because of long-running repetition of a pattern is not recognized. Much of the information learnt is only as of the instant the data sample was taken to train the AI engine or run the data-analytics algorithm. Thus we find that all our AI related algorithms give more focus to logic rather than data, even though we call it data-driven approach.

So, what is a truly data-based algorithm? A data-based algorithm is one where the data represents logic. The presence of data in a certain format automatically creates or realizes a logic because of adding an observer. For example, if take a data series “1, 2, 3, 4, 5”, we know the logic of these numbers as an incremental series, incremented by 1 at each point. If we take a data series “A, C, E, G”, we know the logic of these alphabets as alternate alphabets of the English language. In this manner, in a data-based algorithm, the data represents the logic, rather than logic represent the data.

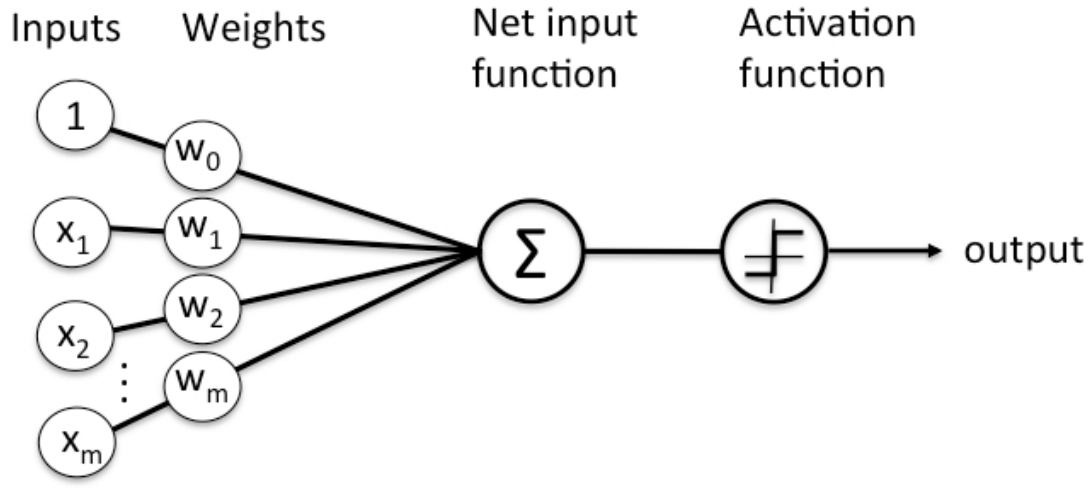
To understand the difference, let’s take the first series and see what happens in both the systems. In a logic-based system, where logic represents data. So, for the first series we write the logic as “next x = x + 1”. Here, we have tied the series down to the incremental series, incrementing by 1. In a data-based system since data represents logic, for the first series, we leave it at the data “1, 2, 3, 4, 5”, adding an observer that can observe the logic “next x = x + 1”. To be noted that the logic need not be represented as a mathematical function, in fact it is not represented as a mathematical function. I have only used it to explain the difference.

Now, say the next number that came into the data series was 2, rather than 6, then the logic we have does not match this data. In a logic-based system “2” becomes an outlier. In a data-based since we have left the data as is it now becomes “1, 2, 3, 4, 5, 2”. Given that “2” does not suit the logic of the previous observer, another observer is added that starts observing from “2” for any pattern that can form a logic. If, as more data comes in a pattern builds around “2”, then two observers recognizing different patterns are present. If not, only the original observer is present and the second observer fades away. This is a “data-based algorithm”. We see that in a “data-based algorithm”, data represents the logic in real-time from data and subsequently only data that match a logic is clustered to a specific observer. Given data drives the logic, data is not discarded. Just more than one logic gets formed, forming multiple clusters as the data come in. All data typically fall in one or the other bucket without outliers.

In the inner workings of reality, the approach used to represent the data, effects an algorithm i.e., a logic is automatically realized by merely introducing an observer of the data. I have explained how this is done automatically because of the way the data is represented in the inner workings of reality in the next section. So, what is this algorithm that is realized by the data representation and how is it realized automatically?

**The presence of a highly efficient, continuous, real-time foundation method**

A foundation method can be thought of as similar to the basic guiding principle that is applied throughout the algorithm. For example, in a neural network, the basic guiding principle is that there are many interconnected nodes. Each node combines inputs from data with coefficients, weights and other "stimuli" to amplify or dampen the input data. This is amplified / dampened input data summation is then passed through the node's activation function to determine if the data needs to progress or not to the next layer and to what extent it needs to progress in the network. The output of the node is then fed as input data to the next layer of nodes in the network which operate on the same guiding principle. The co-efficients, weights and activation functions are trained during the training phase. The guiding principle is typically shown as below:




Example guiding principle for neural networks

The formation of anything in reality is because of a primary foundation method, automatically realized due to the approach used for the representation of data in the inner workings of reality. This simple foundation method is applied and reapplied in various

contexts to achieve varied results. To understand this foundation method present in the inner workings of reality, we need to examine our own reactions to situations first. Then question and understand how it is different from the algorithms we write. From there we understand how we do it and then can project that understanding to the actual method.

As I have said in the previous chapter, we as living beings are able to go into a totally new situation, with little or no experience in that particular situation, react to anything that occurs in that situation to the best of our capability, and as we react to the situation, in-parallel learn from it and adapt it for a similar situation next time. Thus, we keep learning as things occur. We do not need terabytes of pre-collected data and be pre-trained to be able to react to any situation. Sure, to react in a manner that we judge as “perfect” to a situation, we do train, that makes our actions “perfect” according to us. But, it does not imply we cannot react to any situation thrown at us, unlike the AI that we write which cannot react to unknown situations thrown at it. So, what makes us be able to adapt to any situation, while the AI systems that we write only react to those particular situations for which we have trained the AI to react to?

A number of differences in the way AI is coded vs the inner workings of reality go into driving the difference in reaction to an unknown situation. One of the main difference that contributes to it is in the way analysis is done on data between the algorithms that we write for AI and the inner workings of reality. In reality we are able to execute our algorithms in real-time against “truly” real-time streaming data. To be noted that I have called it “truly” streaming data which is different from what is coded as streams say in a language like java. In java, for example, data in an ArrayList can be handled as a stream of data on which “collection algorithms” such as data filtering can be applied. All this implies is, for the data that is already present in the ArrayList, a filter condition is applied on the elements one by one, to create another list of the filtered data, filtered by the condition. It could have very easily been written with a loop syntax to filter the same data.

To give an example of streaming data, real-time algorithms and offline algorithms: Say, I created a program to process the feed from a serial COM port connected to a RS232 thermometer that is continuously feeding in the temperature every minute. The program needs to compute average temperature every hour. The data that is coming into the program is truly streamed in real-time, because the thermometer is streaming the data by measuring the real-time

temperature, as opposed to an already known list or set of data. To process this, the program can either use a real-time algorithm or store hourly data and do an offline computation.

The offline computation is easy. As the data comes in, store the data into a table in the database. Have a timer based thread that wakes up every hour, picks up all the data of the previous hour and compute the statistical average. The problem with such an implementation is it begs the questions: “what is considered as an hour?”, “should the timer run be based on the actual local time of the computer such as at 9:00, 10:00 and so on or should it be a continuous moving one hour average?”. How can a moving average be computed in such an implementation? This can also be done if the timer thread is set to run at a higher frequency of say every 1 minute or 30secs. Every 1minute, the thread wakes up and computes the statistical average of the previous 60 readings. It should be noted that such algorithms can introduced time-based or synchronization errors. It is quite possible that the thermometer has not streamed in the latest reading when the thread wakes up. This implies that we could potentially have the same moving average between two runs of the thread, just because it ran before the thermometer’s reading was stored in the database.

A real time algorithm is better in use cases such as this, where we are looking at a moving average computation. A real-time algorithm works thus: Maintain a moving sum of the previous 60 readings. When the next value comes in, knock off the earliest $60^{th}$ reading, add the current reading to the moving sum and compute the current average. Thus, as the latest reading comes in, the average is continuously computed to get the moving average. Obviously in such an algorithm, the first average will only be available when the first 60 readings have come in. There are many ways in which to handle the moving average for the first 60 reading in real-time easily. This is similar to us handling a totally new situation, where we react based on our experiences from different other situations.

It is seen here that, given that we have designed our computers for mathematical computations, it is easier to write a real-time algorithm for statistical functions rather than any other logic such as knowledge gathering or intelligence computation from knowledge. Further it is also be seen that given ours is a discrete data representation, we are forced to sample and get real-time data at regular intervals, rather than a continuous stream of data that is present in the inner workings of reality.

As I have explained before, given that data directly represents the logic, the continuity of the stream of data in the inner workings of reality allows a creation of a continuous real-time algorithm that becomes the foundation of forming and sustaining reality. This is best understood by the saying “reality is a unerasable drawing”. In reality, data is a stream of continuous real-time data using which we keep drawing, adapting and fine-tuning it as more and more data streams-in.

This is different from looking at data as one huge span of discrete data sets that gets processed once and only once and subsequently, only the logic learnt is used. Our typical AI algorithms are split into phases. The first phase being the learning phase and the subsequent phase applying the learnt knowledge for an application such as predicting or forecasting etc. While data analysis algorithms are written to continuously upgrade themselves from the data collected, they still execute as offline algorithms where the knowledge from the latest data collected is run in a thread in the background as big-data analytic jobs. The knowledge gathered is updated back to be used in subsequent runs. In our world of computers, real-time data analysis is typically limited to simple mathematical functions such as statistics, everything else is forced to be an offline analysis because of the limitations we have in data representation and computing.

The next difference that contributes to the way reactions differ between reality and coded AI is in the data representation on which the algorithm is run. As I have explained in depth in the “data representation” and the subsequent “inherent states of data” sections, the data formats and the inherent capabilities of data that is present for analysis in reality is much more advantageous to knowledge and intelligence systems. The algorithms that we can run on such data is more oriented towards collecting knowledge which is approximate or fuzzy logic as opposed to accurate mathematical computation logic. In our AI systems, we are obsessed with accuracy and try to compute values even to many decimal places. This is because we are obsessed with discrete mathematical representations rather than the actual working of the algorithm. For example, if I want to change lanes while driving a car, I do not care if I am are driving at 60kmph and the car behind me is driving at 55kmph. All I care about is the approximate time taken for my car to switch from the current moving space I am occupying to occupy the moving space in front of the car behind me. I am just answering the question “is that time sufficient or not for the car behind me to catch up and will it bang

into my car". All I care about is, once I have switched lanes, does the speed of my car and the speed of the car that will be in front of me match, so that I will not bang into the car that comes in front of me. Here "approximately" is all that matters, who cares if the exact time is 5.5s or 5.3s or even 5s or the speed is 50kmph or 51kmph or 52kmph? The relative-ness is all that matters. The problem with our computer programs is even though I want rough answers, I am forced to accurately compute values, which leads to wasted compute cycles and obviously leads to accurate cut-offs and comparisons which takes me nowhere.

If we look at the algorithm that is present in the inner workings of reality, we find that they are independent of "**what actually the Brahman is**". This is because it is possible to write an algorithm that is completely independent of the actual something that is an accurate "value of data" and yet be able to derive knowledge and create intelligence out of it.

As I have explained before, rather than looking at the actual property type and the absolute accurate value associated with that data, if we looked at the relative change across multiple data of the same property type, as it is streaming, then, we have decoupled the algorithm from the actual value. We can compare the relative changes of this across multiple properties find relational patterns. So, in the car example above, rather than looking at the value of the absolute speed of the car beside us, we look at "how fast the car beside us is travelling w.r.t ourselves"? Here again the absolute accurate value adds no value to be computed. We need to compute just w.r.t ourselves. So, all we are doing is deciding, "is it travelling faster than us, slower than us or same speed as us?" The way we compute is just to see over a sample period, is the car gaining on us or are we gaining on the car based on the relative positions of the cars. We are trying to compute how long will it take for me to clear the car's front or the car's rear. Again, we do not look for accuracy here, we look for approximate. Accurate time and accurate speed differences make no sense computing and takes a long time doing so. We are in real-time assessing and re-assessing continuously, the moving duration to clear the front or the rear of the car next to us, with real-time stream of data as the car beside is travelling along with us. This is a visible direct application of the base foundation method that is used in the inner workings of reality. So, how do we do this?

As I have described in the description of "data representation", the data representations that is present in this reality, is tuned to

easily do an integral of derivatives of various properties, compare changes of one property against multiple other property changes and accumulate them appropriately to identify patterns, compare those patterns and identify in an infinite recursion patterns within and over patterns. This is not done as a mathematical calculation, but as a collection of the actual data points rather than unifying them into some smoothened regular mathematical function. Both BHPS and the Surya Siddanta, explain how this is done explicitly. But, it is clearly brought out in chapter 2 of BHPS, where it explains about how appearance is rendered. It says the following:

*rAmakRShNAdayo ye hyavatArA ramApateH*

*teapi jIvAMshasaMyuktAH kiMvA brUhi munishvara ||2.1||*

*rAmaH kRShNashca bho vipra nRsiMhaH sUkarastathA*

*ete pUrNAvatArAshca hyanye jIvAMshakAnvitAH ||2.2||*

*avatArANyanekAni hyajasya paramAtmanaH*

*jIvAnAM karmaphalado graharupI janArdanaH ||2.3||*

*Translates to*

*These appearances of rear appearing to stand still are unrelentingly stationary and at-rest. The individual parts there that are moving being inter-connected invokes moving impulses. Along with stationary at-rest penetration, there exists agitated becomes of the individuals at the resting place that generate work, approaching an appearance for the entire penetration and other linked individual parts in the rear. Unifying the other appearances in the weakened rear of the primary nature, the result of the work of the individual acquiring nature of the produced agitation.*

This when read without context seems pretty confusing. Giving a context to this, namely contextualizing “appearances” with a meaning, which is: appearances are “changes” in a stream of real-time data, change of the currently streaming property value from the previously streamed property values. Now, when we start thinking about “rear”, we find that it is a reference to an already streamed value. We need to ask ourselves, when will an already streamed value seem to remain stationary and at-rest when we look at continuous divisions of continuously streamed data?

For example, say we streamed the following data stream (I will take discrete data first): 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4… and so on. To this series say, I define a division

of data as "10-digits" per division, i.e., I observe 10-digits simultaneously at a time, then it will look as below:

| | | |
|---|---|---|
| 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 |

The center is the window defined. It appears to stand still, since this series passes through that window "10 at a time", and this is a repeating series of the same 10 digits over and over again. This is a very simple, regular series. Assume we pick up a window of a matrix of 5X5. Now a sample of this is as below:

| | | |
|---|---|---|
| 01, 02, 03, 04, 05 | 10, 11, 34, 56, 89 | 09, 08, 10, 45, 78 |
| 06, 07, 08, 09, 10 | 06, 07, 08, 09, 10 | 06, 07, 08, 09, 10 |
| 03, 04, 05, 06, 07 | 03, 04, 05, 06, 07 | 03, 04, 05, 06, 07 |
| 08, 09, 10, 11, 12 | 11, 12, 98, 66, 78 | 01, 02, 54, 67, 87 |
| 07, 08, 09, 10, 11 | 45, 09, 10, 87, 54 | 65, 45, 32, 45, 11 |

Here we have the 2nd and 3rd rows repeating. Thus as the data series flows through the window, it appears as if the 2nd and 3rd rows remain at stand still. But the other rows appear to be an agitated flow of some random numbers.

Now assume that instead of looking at this window as directly the numbers, we look at it as a difference of the window data between the first, the second, the third and so on. We would find then that the 2nd and 3rd row will be at 0 always and the other rows based on the difference would give us a certain picture that is an integral of all these numbers differentiated.

| | | | |
|---|---|---|---|
| 09, 09, 31, 52, 84 | 01, 03, 24, 11, 11 | 09, 08, 10, 45, 78 | 00, 00, 00, 00, 00 |
| 00, 00, 00, 00, 00 | 00, 00, 00, 00, 00 | 06, 07, 08, 09, 10 | 00, 00, 00, 00, 00 |
| 00, 00, 00, 00, 00 | 00, 00, 00, 00, 00 | 03, 04, 05, 06, 07 | 00, 00, 00, 00, 00 |
| 03, 03, 88, 55, 66 | 10, 10, 56, 01, 09 | 01, 02, 54, 67, 87 | 00, 00, 00, 00, 00 |
| 38, 01, 01, 77, 43 | 20, 36, 22, 42, 43 | 65, 45, 32, 45, 11 | 00, 00, 00, 00, 00 |

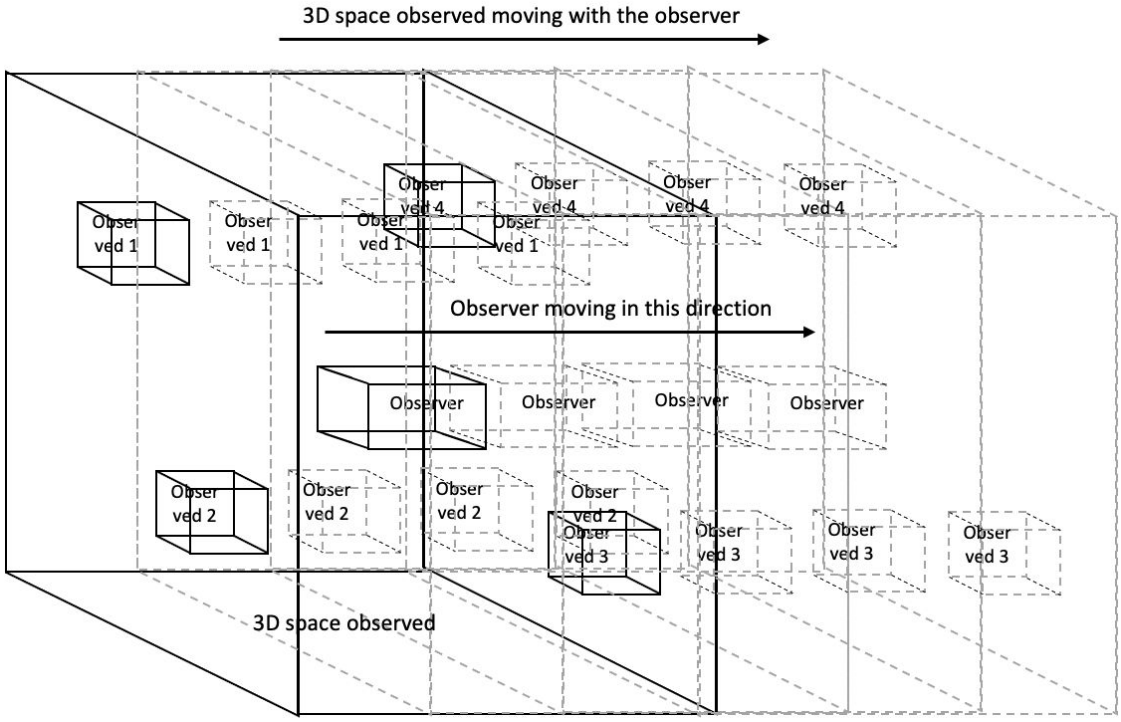
The matrix generated by the window now becomes a differential or rate of change of the stream of data represented by the actual data rather than some regular function that we always try to represent as, in mathematics. For example, when we look at a function $f(x) = x^2$, the derivative function of this is $fd(x) = 2x$. This can be represented in this manner as a regular function or can be plainly represented as data. So, f(x) can be represented as the data series: 1, 4, 9, 16, 25,

36, 49, 64, 81, 100…. The delta of this is 1, 3, 5, 7, 9, 11, 13, 15, 17, 19…. The rate of change is (1+3)/2, (5+3)/2, (7+5)/2, (9+7)/2, (11+9)/2, (13+11)/2, (15+13)/2, (17+15)/2, (19+17)/2… which is the derivative of this function and can be represented by the data series: 2, 4, 6, 8, 10, 12, 14, 16, 18…. Here, instead of representing the function as a regular function varying with a variable represented as 'x', we have represented it as just the streamed data, as I explained previously in data-based algorithms. Why would we want to do this?

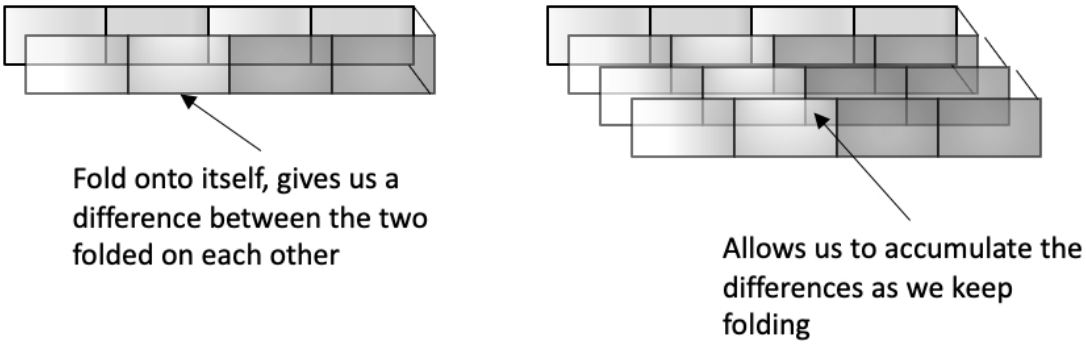
This is useful when we have a irregular data series that does not vary by any set of known relatable variables and cannot be represented by regular smoothened functions. When we look at reality, we find this to be always the case. *We do not want to regularize it just to find a derivative and integral*. Especially when we are looking at real-time algorithms, adding more steps to process just adds to the complexity.

Given that derivatives are just rate of change at any point from the previous point, delta can be computed as "difference from the previous point" and derivative series can be computed as the "average of difference of the current and previous delta. Accumulate these derivative data series and we get the integrals represented by data directly. An even more important application is when, rather than doing the rate of change computation at every single data point against the previous point, we split the stream into chunks of windows as I have done above with matrix and compute rate of change against data in the corresponding positions, i.e., second position in window again second position in the previous window. In fact if we observe ourselves, we are only doing such a integral of differentials, we rarely do the regular previous point differential as we do in mathematics. Why do I say this?

Let's take even the simplest act of vision. We see a scene, which is a 3D continuous visible space and are able to in-parallel see the change across the 3D continuous space. We do not do a sequential processing in that space, but do a parallel processing. In that 3D space instantaneously we are able to see the whole tree swaying. How is this useful? If we took the example of driving a car again: we typically know the 3D space around our car, we compute the differential of that 3D space continuously and look for changes and in that space rather than the retention in that space. Subsequently based on the change, we take a decision as to what the reaction needs to be. The below shows the 3D space and the observer:




The algorithm starts at the 3D space with the solid lines. The data observed is the continuous 3D data space represented as I have indicated in data representation. There is no discrete data here. As the observer in the center moves, the 3D data space moves with the observer as continuously as the observer. Let us say, I do a *differential of the first 3D space with the next continuous 3D space as I had suggested earlier folding over the data or overlaying the previous 3D space with the currently observed 3D space*: If the various "observed" move at the same speed as the observer, I should see no difference in the continuous 3D space. It should remain undisturbed. If one of the "observed" moves faster than me, as I continuously do the differential, I should see a pattern, where the distance between me and the "observed" decreases steadily. Again, to stress the differential is done by continuously overlaying the current 3D space over the previous 3D space. This can be seeing as if a paper is being folded over itself and the two ends moving in opposite directions, like the ribbon of a printer. Observing via both the folded paper, continuous differential is observable by the observer as a continuous data series. And so on, based on the various events happening in the 3D space, the differentials and subsequently the integral as data, should give me the information needed to take a decision. The below diagram shows the differential:


Fold onto itself, gives us a difference between the two folded on each other
Allows us to accumulate the differences as we keep folding

To note, I have only used only properties related to “vision” here to identify changes. The same can be done with sound, with heat and many other observable properties available in the same 3D space observed. The 3D space observed can extend beyond the 3D space associated with vision based on which property is used. For example, sound will increase the observable 3D space while heat will decrease the observable 3D space. Thus, we see that a very simple continuous real-time algorithm implemented with the correct data representation can give us abundant knowledge and information without much processing.

It should also be noted that there are many advantages to the above type of processing. As I have said before, since, we are dealing with continuous stream of data in 3D space and the algorithm is a simple overlay of one division over the previous, the processing is also parallel and done in the 3D space instantaneously and as continuously as the observation of the 3D space.

While the above is just an example using space and observation in space as an example, it equally applies to any data representation in reality that can be interpreted as a series data or a continuous data. So, we are not limited by the dimensions of space. We are talking about any data in a vast expanse of “the Brahman” which is dimension-less and before any manifestation or meaning or rendering has been attached to it, that can be observed. Thus, when we talk about simultaneous instantaneous processing, we are talking irrespective of the rendering. Space or time dimensions has no meaning to such a data, unless the data has a property that is specifically observed as a space dimension.

So, what happens to data that is “way off” from the other data that we observe in such a realization of integral of differentials. If we are dealing with numbers that are discrete, we can omit the outliers to the regular function and look at it as a smoothened regular function varying on variables. But, with continuous bits, what is called an outlier is very difficult to compute, because discrete points

are not present. Moreover outlier is just a concept in the discrete, regular function world. If we are looking at continuous data being the driver of everything, then, there are no outliers.

When we look at the series of data as continuous bits and keep folding or overlaying them over each other, we find that the deltas and the differentials automatically weed out the outliers. We do not need to add algorithm separately for this. This is because a data that is a one-off occurrence gets automatically phased out as more changes get overlayed over it and there does not exist changes that reinforce the outlier, effectively removing outliers.

This is also explained in the subsequent verses in BHPS:

*daityAnAM balanAshAya devAnAM balabRddhaye*

*dharmasaMstApanArthAya grahAjjAtAH shubhAH kramAt ||2.4||*

*Translates to*

*By the contraction of the inactive strength, the strength of energy expands causing order to be established, acquiring that which propels the production of suitable series.*

As I have explained in the section “data representation”, we have dormant data and activated data. At a particular region as the dormant regions contract, the regions with activated data or energy expands. This gives us many continuous bits of activated data that can be divided and overlayed on each other. This gives us better results on the series. While we can work with little or no data, the more the data, the more accurate the result is.

So, what is suitable series? This is where the outliers are eliminated. Only those series of patterns that repeat or occur consistently will be observable. Anything else will be weeded out. Further verses tell us this:

*rAmoavatAraH sUryasya candrasya yadunAyakaH*

*nRsiMho bhUmiputrasya buddhaH somasutasya ca ||2.5||*

*vAmano vibudhejyasya bhArgavo bhArgavasya ca*

*kUrmo bhAskaraputrasya saiMhikeyasya sUkaraH ||2.6||*

*Translates to*

*The transformed of the pattern of the stationary appearances descends from the principal at-rest, by awakening the individual resting places in the successive base along with extracting the*

*related transformed. By controlling the expansion of that thought to a first thought and from that first thought work done radiates into the successive similar resting places that generate work.*

When we take a continuous bit of activated data and overlay one division over the other, as I have said before we get differentials. But, this is because, "*we"* look at it as differentials because we have learnt and defined it in mathematics. If we did not have mathematics background and had not known about derivatives and integrals, what would we see?

This is explained in the verse above. We would see "stationary patterns" at a specific region (in the matrix example above the 2nd and 3rd row) of the division or window which keeps getting overlayed with the same data and the other regions will appear to be randomly changing. Thus, many individual regions which remain stationary patterns appear based on how many regions have little or no-changing activated data.

If we look at what we have done, we have picked up activated data (not numbers, but continuous changing thoughts) computed differentials and accumulated them, which gives us what we call "integrals" in mathematics. And this is automatically split into recognizable patterns. Thus, we are able to very efficiently do the mathematics of "integral of differentials". But, what we further go ahead and do is "recognize and divide that result of integrals into patterns", repeating patterns, non-repeating patterns easily, which is what we try to do in clustering logic in our AI with complex logic. All this, just by changing the data representation and adding an observer that can view through folds or overlays. To do this with discrete data and a computation based processing, we will have to write a hugely complex code.

Further, these verses tell us, if we can "control" the appearances of these "stationary patterns" from the first thought and keep continuously radiate out the activated data to form the subsequent thoughts that would maintain that first thought, then we have done *work[more about work in the next chapter]*.

It should be recognized that given that the inner workings of reality is in this manner, we find that the data representation that we work with has naturally 4-dimensions.

The first dimension is "uncertainty" of data. Given that we work with a change or delta from the previous values, it already represents the "*uncertainty*" of the occurrence of the data. This is determined

by using the density of the accumulation. The density has an inverse relation to uncertainty, hence if the density of the accumulation is high, uncertainty is low and if the density of accumulation is low, uncertainty is high. If the uncertainty is at zero, then it becomes an outlier and is weeded out.

When we work with change in data, we cannot compute changes unless there is a reference point from which the change is computed. This is the "*origin*" of the series. The existence of such an origin allows us to infer a "*location or distance*" information for the data automatically. This is the second dimension. To be noted, that the distance talked about here should not be mixed with "distance in space" that we usually envision when talking about distance. This is "distance of data" and depends on the computation that we do. For example, "distance" in a NLP algorithm is typically computed as the number of words in-between the two occurrences of the words for which distance is computed. In this case, if we had a stationary pattern of sine wave that is repeating, then "distance" can be equivalent to the wavelength or distance can be equivalent to "amplitude". Thus, the representation of distance varies easily based on the context from which distance is computed.

Since we are computing change, from the origin or some previous data, we know if the current data is a positive change or a negative change. This allows us to know the direction of change. This is the third dimension. "Direction" again like distance has varied meaning and should not be confused with the "space direction" that we usually think about when "direction" is mentioned. As I have explained above, "positive" and "negative" change is one type of direction that can be computed. But, to be remembered, positive and negative also has no meaning in this context. It needs a second parameter based on which it can be tagged positive or negative. This is called the "direction of the movement of path" which is termed as the "forward direction" and any other w.r.t that forward is a reverse direction from the forward direction.

Thus, purely because of the way the data is modelled and the algorithm applied, we have the four dimensions associated with any data: "change", "uncertainty", "distance" and "direction". Also, from the way data is represented, we already have the quadraple (observer, type, value, state).

**The compose-decompose philosophy**

There are many advantages to representing data in this manner and realizing the algorithm in this manner. Note, I have called it

realizing the algorithm rather than executing, because there is no execution. Purely because of the manner of observation of the data, logic has automatically formed. One of the major advantages of such a realization is what I call “the compose-decompose philosophy”. As we see from the realization of the logic, based on the amount of data folded upon each other, the observability is changed. It is so modular in nature that more continuous bits of activated data can be overlayed one on top of other to it or continuous bits of activated data currently overlayed removed from it and the logic will work as expected for the data that is current overlayed. This philosophy is the next fundamental concept that is present everywhere and highly used across all places in the inner workings of reality.

A best example of a “compose-decompose modular logic” in the computer world is a recursive function. For example, say the following is the pseudo code for creating a fibonacci series:

```
public int printSeries(start, current, currcount, max) {
        System.out.print(start + “,”);
        if (currcount <= max) {
           printSeries(current, start + current, ++currcount, max);
          }
}
```

For example, calling this function as printSeries(1, 1, 0, 10), will print the first 10 digits of the Fibonacci series. When we look at the execution of this logic, the top-level call encompasses the whole set of 10 calls to the printSeries function, which is a composition of the same logic, but can be decomposed into pieces of logic which is the same as the individual piece. It should be recognized that in the recursive example I have given, it is discrete blocks of logic that needs to be composed and decomposed. This is because computers are discrete in nature. But, given that reality works with continuous bits of data, the concept described here is applied on continuity as easily. The continuity is such that at any point the split can occur and till any point accumulation can occur.

We find that the concept that is brought out in this type of “compose-decompose modular” logic is a fundamental philosophy that is followed over and over again all through the algorithm, applied at every turn for a different concept. In the above it is applied to the data representation realizing a fundamental method of creating logic. Split the continuous bits at any point or accumulate more continuous bits till any point, it still will generate a completely valid logic. This same philosophy can be applied to data, where data

can combine to compose a continuous bit or whole or decompose into bits and still be completely valid data. Later on, we will find that it is the same logic applies to the fundamental component i.e., cycles, the same can be applied to the goal of the algorithm and so on. This philosophy aptly described in the following mantra in Brhadaranyaka Upanishad:

*Om PUrNam-Adah PUrNam-Idam PUrNAt-PUrNam-Udacyate*
|
*PUrNasya PUrNa-mAdAya PUrNam-Eva-AvashiShyate* ||

*Om Shaantih Shaantih Shaantih* ||

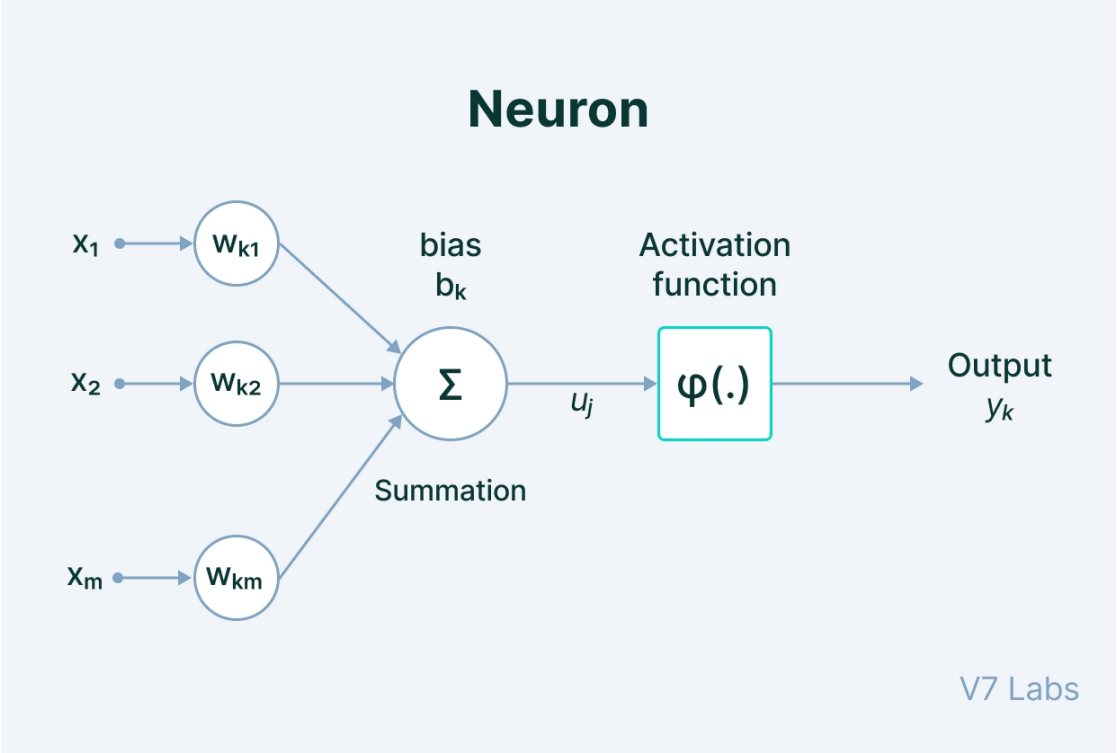
*Translates to*

*The OM of that whole is a whole by itself and that whole of the whole renders, If you take that whole and split it, that which remains is also whole and what is split is also whole and this is OM in shanti*

While, this mantra is typically translated with PUrNa translated as “a whole or entire”, in fact this applies to many contexts and applies aptly when asked “whole or entire of what”. In this case it is either data, logic, cycle or goal. Every one of them follow this philosophy. Truly I find, had we used this philosophy to create our technology, we might not have gone wrong with them. As such, when we look at what we have created as technology currently, it more and more seems like we have successfully ended creating garbage, because that is where everything ends up.

**Fundamental architectural components**

Fundamental architectural components in any software application are the underlying framework components that can be combined in many different ways to achieve the goal of the application. In a neural network, a neuron forms the primary component that is repeatedly connected to various other neurons to create a network of neurons. The key components of the neuron are as below:




Here, the inputs are weighted using various weights which are then passed through a summation which then passes through an activation function to either activate the current node. As I have indicated before, typically the activation function which is a mathematical function, along with the weights and bias are learnt as a part of the learning algorithm.

A number of neurons come together into a single layer and multiple layers connected in a graph forms the network. The first layer is the input layer, the last is the output layer and there are many intermediate layers. When the layers are 10 or more they become a deep neural network. Based on the number of layers, the feedback of the output of the neurons at different layers and the processing that is done at each layer, the type of neural networks vary and achieve various functions. Irrespective of what the over all goal of the neural network is, the goal of the "neuron" itself is to combine the various inputs in varying weighted fashion and shift the output by giving prominence or bias to one or more inputs so that the subsequent redirection into the next layer of neurons can be done correctly and thus give the correct over all output for the neural network. It should be noted, that in neural networks, once the learning is completed, the neurons have no way of being controlled in anyway except via the input data.

Is there such a component similar to a neuron, in the inner workings of reality? What is goal of such a component? What are

the inputs, what are the outputs and how is the component controlled? In a logic-based system, logical components are present and has a static composition of logical constructs whose execution vary based on data provided as input. But in a data-based system, rather than logical components, we will have components defined by divisions of data that they work with. These components are defined as aggregates of data that is divided based on some “abstract framework”.

In a logic-based world, we have a “logical framework”. When we look at the computer applications, we call these “technical frameworks” which is used across logical applications irrespective of what the application achieves. For example, an “event processing framework” is one such technical framework. This allows any application to use the framework as a publisher and / or as a subscriber. As a publisher an application can create events and post them into a topic on the queue, while a subscriber application uses it to subscribe to an event in a topic and define the logic to be invoked when an event is available. As the applications of computers are logic-based, these frameworks contain common logic that remains the same across any application using these applications.

When we look at a data-based system, the only common logic that can be present is a logic to traverse and divide data into components. In a data-based system, this logic is typically automatically realized because of the way data is represented. As I have indicated previously, these data components are divided into components because of the visibility of data. According to the Surya Siddanta from this visible data, logic is formed and that logic is used to progress further. So, unlike the technical frameworks, the abstract framework here is an abstract logic that guides the data through various phases rather than a logic that is run on the data.

The abstract framework that works with data within the field of vision of an component, establishes what I call the “cycles of existence” or as it is called in Sanskrit, “piNDa”, which literally translates to “spheres”. Surya Siddanta talks about what are called “jyApiNDa” which translates to “controlled spheres” or “controlled cycles of existence”. It should be noted, that these cycles of existence use the real-time foundation method to achieve their purpose on the data falling within the field of vision of that cycle of existence. It should be noted that just as how technical frameworks don’t run by themselves and are driven by the applications using them, these cycles of existence are also not driven by themselves,

but by the overall algorithm that drives the inner workings of reality. They cannot exist by themselves, isolated from the algorithm.

To understand these cycles of existence, we just need to observe the natural paths that are present in nature. These are the transformed versions of underlying cycles. For example, the most famous cycle that is taught in every primary school is the rain cycle. In this cycle water from rivers, oceans, ponds and every other water source on the ground evaporates to form clouds, which then precipitates as rain to fill this very same water sources on the ground and replenishing the ground water. Again, while this cycle is taught as a natural cycle for water replenishment, it has been unrecoverably altered by human intervention, when lots of water sources are filled with material that do not absorb water, causing irreparable damage to the cycle. Another very common example is the growth of trees, which is not as famous and is not taught as a cycle, but is a cycle all the same, had it not been intervened by humans. We can start the cycle at the seed being sowed in the soil with nutrients, which sprouts and grows with the nutrients in the soil, in the fall season the leaves and flowers fall, these are soaked during monsoon and dried by the summer to become mulch and fertilizers that form the nutrients for the soil into which the now ripe fruits fall to decompose, except the seed, that seed now grows to a plant completing the cycle. Obviously given that humans interfere with this cycle, by either preventing the natural fall of the leaves by unseasonal adding of nutrients, or by cleaning out the decomposing leaves, this cycle has also changed to unrecoverable levels.

Even within these visible cycles we can see many cycles interacting with each other and being composed of other cycles. It should be realized that these cycles of existence go all the way to the depth of truth from where the appearance that we observe as reality come and have been translated with a kernel of the internals of working of the reality present in it. While, it maybe difficult to wrap our minds around the fact that these natural cycles forms a logic in the underlying working, it is such cycles that form the framework component of the inner workings of reality.

As can be readily seen, there are many differences between the way these cycles of existence work and the neurons of the neural network work. The first being these cycles are formed around data rather than logic and hence each and every cycle of existence has a field of vision in the continuous whole within which it operates. It should be understood that the data in the field of visions of each of these component cycles can be aggregated to get back that

continuous whole from which the data for the fields of vision was carved out. Hence it follows that the logic that works on these smaller pieces of aggregates of data can be combined together to get the higher logic that occurs on the aggregates of the data in the smaller pieces that composes them and so on till the continuous whole is got back. Thus, all logic that is learnt by such a system can be looked at as following the "compose-decompose philosophy", where any "piece of logic" in the system can be decomposed into modularly similar logical pieces or multiple pieces of logic in the system can be combined to form the higher logic. Both the data and the logic follows the "compose-decompose" philosophy as I have explained.

Thus, when we take the topmost level cycle of existence that is the root component of the algorithm, it is composed of every other "cycle of existence" and recursively composed of sub cycles of existence. Each and every cycle of existence from the topmost level to the lowest level are controlled by a single control that translates across all the cycles of existence and achieves the same purpose within the field of vision for that cycle of existence. What should be noted is that it starts with data and creates logic out of the pieces of data. When all the logic is aggregated we get the final top-level logic.

**Equilibrium as a abstract recursive goal**

As I have described previously, in a neural network, irrespective of the goal of the neural network, the goal of the neuron is to combine the various inputs in varying weighted fashion and shift the output by giving prominence or bias to one or more inputs so that the subsequent redirection into the next layer of neurons can be done correctly and thus give the correct over all output for the neural network. We see here that the abstract defined goal for the neuron is in no way related to the overall goal of the network. This is a major difference between the algorithm of the AI that we code and the algorithm of the inner workings of reality. In the algorithm forming reality, the continuity of the algorithm is present in every dimension and direction. As I have indicated previously, we miss the key component of continuity in our algorithms, whether it be continuity of data or algorithm. We all understand what data continuity means because we have seen the analog nature of space and time. Divide space or time into any number of discrete parts, you can take each divided part and divide it further into even smaller parts without losing the properties that make up space and time. But, what does continuity in algorithm mean?

To explain, let me take an example. If we take a glass of juice with fruit chunks in it and stir it so that it creates agitation in the liquid. Then if we let it stay, it will try to come back to rest. What is to be seen here is that all the fruit chunks in the fluid will also try to come back to rest and the rest state of the fluid is affected by the rest state of the fruit chunks. Without the fruit chunks also coming to rest, the whole fluid will not be at rest. Each and every molecule in the fluid as well as the molecules in the chunks also try to come back to rest. Thus, we find that the logic to follow to come to rest has been transmitted all the way from the macro level of all the fluid in the glass, to the chunks of fruits to the molecules within the fluid and the molecules of the chunks of fruit. Only when all the continuous sub-components have the forces are balanced will the top level juice have its forces balanced. We need to realize that not all of them need to follow the same path or have the same time to come to rest, but at the end when the whole liquid is at rest, every other micro level elements are also at rest. This is the same as the compose-decompose philosophy of everything else in reality. This is continuity of algorithm, where the higher algorithm translates to an aggregation of continuously dividing sub-components below.

One of the reasons why such an algorithm works is because the higher goal of coming to rest is also transmitted all the way through, from the macro to the micro levels. This is visible visually in fluids that are agitated and trying to come back to rest.

The data in the inner workings of reality can also be seen as a continuously, perpetually churn of data. With every change the system churns to adjust itself to the changed state, which introduces another change and the cycle continues. Here, the data is trying to come to state of some defined equilibrium. This is the top-most goal of the algorithm and is transmitted all the way from there to the macro components and from there to all the micro components similar to the agitated juice. So, while in neural networks, the components i.e., neurons have a totally unrelated goal, but go forward to achieve a specific network goal for a specific input, that is not the case with components that form the inner workings of reality. All components formed in the inner workings of reality from top-most to the macro to the micro components take on the same goal of “coming to equilibrium”. If the micro components can be divided further, they will also take on the same goal. Hence, the continuity of the algorithm is present in every dimension and every direction.

# The algorithm of the internals of working of reality

In the previous chapters I explained the data representation, the states of the data, the components, the guiding philosophy, the foundation algorithm, the foundation component and the other guiding principles in the algorithm. It is time to tie all of these together into the actual algorithm that runs to create this reality around us. As I have described before in evolution, we need to look at this algorithm as that which allows something that starts at a point where there is just a small change between the surrounding environment and itself, from there that small variation grows in an ever extending, expanding algorithm, that keeps bringing in more activated data into the fold till the top-most cycle of active-inactive is established.

Physics tells us that perpetual machines are not possible, and every machine needs to degrade over time and fade away. But, if we think about it, that is only within the reality. But, to have reality itself existing continuously around us, we need it to be perpetually repeating and expanding in order for this to exist stably. So, this reality around us should in fact be a perpetual, open thermal system. As I have translated before, the Nasadya Suktam of Rigveda tells this in this manner:

*tama AsIt tamasA gULham agre apraketaM salilaM sarvam A idam |*

*tuchchy enA abhva apihitaM yadA AsIt tapas as tan mahinA jAyat aikam || 3 ||*

*kAmas tad agre sam avartatA Adhi manaso retaH prathamaM yadA AsIt |*

*sato bandhum asati nir-avindan hRdi pratIshyA kavayo manIshA || 4 ||*

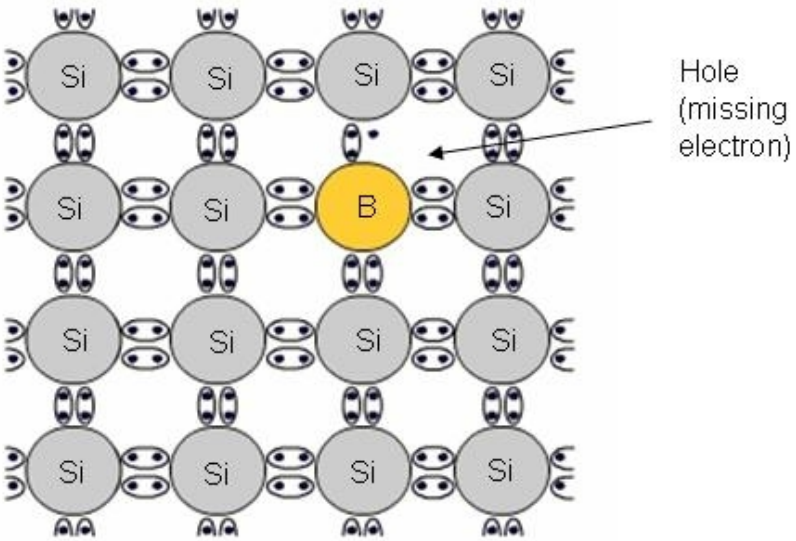
*Translates to*

*Unbound, inert, foremost covered by dormant, all is indistinct, unsteady is this, minute in that enormous enveloped, if unbound heat exists, there in that mighty is produced that one foremost agitation, attached to that, repeating, observing the stream of the thoughts unbound to the first, equally connected, without end, not finding the core, fluidity described as thoughts.*

So, it all starts with some “agitation” that occurs in that vast enormous unmanifested and then repeats without an end, perpetually. While “sam” is translated as “agitation or disturbed”, we need to ask ourselves what agitation really means when we talk about an expanse of “something that is not known”. It should just be a change in an expanse of unchanging environment. Once the cycle is started with the foremost agitation, it becomes self-sustaining. This “change in the expanse is called rajas or dopant” in BHPS.

## Starting at a dopant

Typically rajas is translated as “the quality of passion”. Applied to various context, this takes on various meanings. So, why do I translate it as a “dopant”? In fact what does a dopant mean? Dopant is an electronic term that is used w.r.t to semi-conductors. An intrinsic semi-conductor looks like a lattice of atoms with no free electrons and hence cannot conduct. To this an impurity or dopant can be added to make it a conductor. Depending on the type of dopant either free electrons (-ve hence n-type) are introduced or holes are introduced (+ve hence p-type), making the lattice of the semi-conductor conduct by propagation of the free electrons. So, when silicon is doped with aluminum it becomes a p-type and doped with phosphorous makes it a n-type semi-conductor. For example, shown below, boron dopant:



As we see above, in a lattice of all silicon atoms, the boron atom acts as a dopant that introduces a hole, because an electron is missing. Abstracting this into a concept, the lack of an electron or addition of a free electron can be seen as a change that is introduced in a homogeneous lattice. A similar concept is present in the continuous whole of the Brahman, though it is not a homogeneous whole, but an indeterminate whole. Subsequent to this comes the concept of movement. The below verse in BHPS talks about this:

*vyaktAtmakastribhiryuktaH kathyateanantashaktimAn*

*sattvapradhAnA shrIshaktirbhUshaktishca rajoguNA*

*Translates to*

*By connecting the nature of manifested to the directional becomes, restrains the rendering of the infinite capabilities, the true essence of the natural state is mixing the becomes capabilities and the penetration capabilities and is a quality of dopant*

As per this verse in BHPS, if we looked at the qualities of the dopant, it mixes the capability to become and the capability to penetrate. It should be recognized that a dopant is necessary for penetration which is the manner of data traversal as I have explained in the next section.

## A generic random data traversal technique

I have described in the data representation section the differences in which data is represented in the inner workings of reality and how it compares to the binary representation of data in computers. As I have described, the data representation provides various inherent advantages or introduce disadvantages to the algorithm. The next major necessity of any of our big-data algorithms is a data structure that allows for an optimal method of data traversal. There are many data structures that we have developed and various traversal strategies available for these data structures. But, all of them fall flat when the data is huge and starts balking at a certain data volume unless the data is split into some sort of indexed list or map or group of data structures and then traversed.

AI with neural networks use graphs data structure to store and traverse the various logic learnt. In neural networks, since *the logic* is represented by graphs and not the data, the problem only starts manifesting itself as the number of distinct data sets to be serviced by the neural network increases to a very high number. Graph

traversals varies based on the type of graph created, directed, undirected, fully connected, partially connected, star, cycle, grid and so on. But, whichever the type of graph that is created or data structures that are created, a random access data structure is pretty difficult to create, unless it is a map with a key. But, a map with a key does not store any relation between data. Hence, the data representation and data structures we have in computers do not serve all the required needs. We are forced to represent the same data in multiple formats to achieve different functions.

As I have mentioned many times previously, basically because we started out with computers as purely computing devices with discrete data representation, we are forced to concede that we need to add data structures for data traversal. Given that our data representation is discrete, we have to find ways to related various discrete data in some structural format so that we can traverse the data, taking into consideration these relations between data. This works well for mathematical computation because we know the relations and they are pre-defined using mathematical functions. But, in an AI application, we do not know the relation between data upfront. The whole point of AI is to learn the relation between various data and create a logic out of it.

For such an application, we need a data representation in which relations are automatically present or are easy to form as an algorithm runs. In an AI application, it is more important to traverse all the data rather just a sub-set of discrete data to get the relations. With discrete data representation with explicit definition of relations, traversing all data becomes time consuming. Hence, we try to introduce filters to filter out the data traversed to reduce time taken. But this reduces the effectiveness of the algorithm to collect knowledge out of the data, because we have pre-filtered the data. This implies that we need to code in some logic that is based on expert human intelligence to be able to learn from a data set. When done in this manner, discovery of relations can only happen in the sub-set of data already identified by an expert in the domain. While this may be acceptable for use cases where experts are available, for research use cases this does not fit the need. The whole purpose of the research is to discover relations between various data available.

In the inner workings of reality, the data representation is a continuous whole, where bits can be activated for an observer, as it is traversed, in any direction, by just observing and associating a property and value to it. For such a data representation, “rigid data structures” are not necessary since related data are easily grouped

together by the appropriate application of the foundation method. Also, for such an data representation, data traversal takes on a totally different meaning. The representation of data in computers is finite and discrete, while the data representation in reality is infinite and continuous. The problems of finite and infinite are completely different from each other. What is needed for infinite is not a traversal that "touches" all the discrete nodes in a grouped data structure as is needed in a finite, discrete data set. What is needed for the infinite continuous data is a controlling or limiting factor, that can setup a reference and carve a traversal path and direction from some marked reference in the infinite which typically has no reference. This is what is said in the verse of BHPS:

*vyaktAtmakastribhiryuktaH kathyateanantashaktimAn*

*sattvapradhAnA shrIshaktirbhUshaktishca rajoguNA*

*Translates to*

*By connecting the nature of manifested to the directional becomes, restrains the rendering of the infinite capabilities, the true essence of the natural state is mixing the becomes capabilities and the penetration capabilities and is a quality of dopant*

According to this verse, if the nature of the Brahman is taken and to it is connected some set of "becomes" i.e., "give it direction", then the restriction automatically starts happening to the infinite capabilities. This is similar to setting up the reference. Once a reference is setup, this is similar to climbing a steep mountain, where we need to look for footholds that are present at each step of the way and find the next foothold to go forward. These footholds for the data representation we have are "gaps" in the "continuous bits that are activated". Gaps are those indeterminate or dormant continuous bits in-between activated continuous bits of data. If there are no such gaps present, then there is no way for the continuous bits to expand into. But, even more important is that it cannot know that it is activated. It is similar to a sensor working, there needs to be a comparison and a change in state for any sensor to sense and act. Hence, the BHPS says the following:

*sUrya natvA grahapatiM jagadutpattikAraNam*

*vakShyAmi vedanayanaM yathA brahmamukhAcCautam ||1.6||*

*shAntAya gurubhaktAya sarvadA satyavAdine*

*AstikAya pradatavyaM tataH shreyo hayavApsyati ||1.7||*

*na deyaM parashiShyAya nAstikAya shaThaya vA*

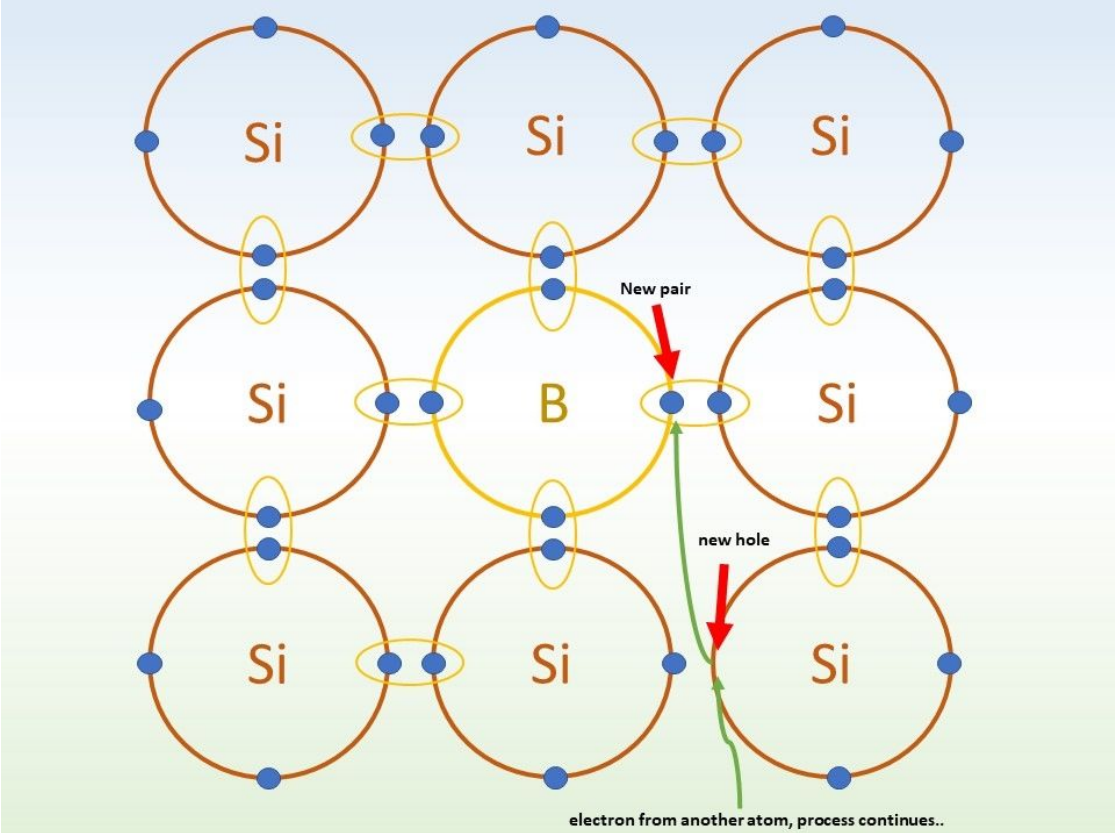
*datte pratidinaM dukhaM jAyate nAtra saMshayaH ||1.8||*

*Translates to*

*The cause for the movable/unmovable is nothing else other than appearance of the acquired pattern, the rendered guided by knowledge of the closely woven direction of creation. The regular distribution of density across the every rendered truth and continuous aggregation of the given energy, facilitates binding of scattered movement. It is not given because of remote guidance nor aggregation of that movement, there is no doubt that, it is produced by the contraction of the gaps between regular divisions.*

Without gaps, there is no place for the activated data to grow into and hence divisions are not formed and we do not have reality. Just as I have said before, the data representation actualizes an algorithm, here too, the data representation allows for easy realization of a method to traverse a path in the data. Given that the data representation is in a continuous whole indeterminate, traversal from any continuous bit of activated data in that continuous whole, is just an "action of penetration" of the indeterminate in any direction. The data should be activated because of the presence of the observer that associates a property and value to the indeterminate. The direction of penetration is determined by the goal of the algorithm, which is to establish an equilibrium.

So, what is penetration? When we read Surya Siddanta or BHPS, we find that this one concept "the act of penetration of the imperishable" comes out in them, which allows to traverse the data. The Sanskrit terminology used for this is "ash" which is typically translated as "pervade", but taking the context into consideration, I find "penetration" is a more apt translation of this word. To explain "penetration" using the semiconductor world again, this is equivalent to the movement of the hole or free electron introduced that makes the semiconductor a conductor, as shown below:




In the above picture it is shown that, as the electron from other atoms shift into the hole introduced by Boron, the hole itself moves in the silicon lattice. If the free electron that is occupying these holes are from a positively doped side of the semiconductor, say by phosphorous, then the perspective from the electron, is that the electron from the phosphorous is moving in the silicon lattice. If, the introduction of an electron or lack of it can be seen as a change as I have explained in the previous section, then this movement of the hole or electron can be seen as a "*change that moves*". In the above only one electron or hole movement is shown. If multiple similar dopants are present, as the change moves, it collects more similar changes and the whole set of changes seems to move within the continuous whole. This can be visualized as an "*act of penetration*" in the continuous whole, because, as it moves the indeterminate is becoming a determinate activated data for that observer. So, it seems as if, *in that indeterminate, unordered continuous whole, an order is being established*. This order established associated for an individual observer is called "dharma" in Sanskrit.

Just as how, what the change is in a semiconductor: a free electron or lack of it, based on how we see the change, what that "change" is in the continuous whole, depends on what the observer "*has detected as change to form the activated data*". Looking at it from the abstract terms, we can decouple the observer and talk about

“this process” as an “abstract algorithm” that occurs to create reality, with just “*change*” and “*penetration of the change*”, because these two concepts are present for all observers.

This “change” in that indeterminate continuous whole is called “rajas or dopant” in BHPS. The Surya Siddanta talks about it as just “change” or “prANa” that penetrates. But the explanation of the formation of the minA rAshi in BHPS, it reads as follows:

*shIrShodayI dyuvIryADayastulaH kRShNo rajoguNI*

*pashcimo bhUcaro ghAtI shUdro madhyatanurdvipAt ||4.15||*

*Translates to*

*The head arises by the clustering of the at-rest dopant quality, by the seeking of strength of interference, destroying the reverse movement of becomes in the middle of the two-part manifestation born due to the tails.*

Here, we see that the “head” of the comet shape formation is started off by the “rajoguNa”. When we look at change or prANa and ask ourselves what will be considered a change, we can see that it has to be a small difference from the surrounding property values. Just like in a silicon lattice, we need to have an impurity so that free electrons can be introduced, even in a substrate of truth that has a lattice of various indeterminate whole, a small detectable change needs to be introduced before the changes can accumulate and form the comet shape. But, what is even more important is “how much of the change is required, for it to become impactful”. The BHPS says this:

*tripAdaM tasya devatya hyamRtaM tattvadaRuianaH*

*vidanti tatpramANaM ca sapradhAnaM tathaikapAt ||1.11||*

*Translates to*

*Its energy has 3 divisions, when the previous division perishes that flows into another, that extent and its natural state are ineffectual in its isolated appearance.*

Just as how, just adding one atom of Boron in a semiconductor is pretty ineffective in what it can achieve, even here, it says that just one isolated appearance of a particular change is very ineffective. It needs to accumulate to an effective level to get manifested. But, the first atom of change is very necessary to start off the accumulation.

Thus, the dopant or rajas as very important requirement to start off the whole process of creation of a thought and reality. While, the

"**first dopant**" is necessary to start off and create the whole cycle, it should be understood, dopants are needed throughout, for subsequently maintaining and keeping the cycle alive also.

## The controllable active-inactive cycle

As I said in the data traversal section, the primary requirement to realize a path for traversal in the indeterminate is a goal or aim. As I have explained previously, while in the AI algorithm we write, the goal varies from one algorithm to the next and within the network created, in the inner workings of reality, there is only one abstract goal namely "equilibrium or Vishnu" or stability, which is the nature of "The Brahman". The BHPS says this about it:

*ekoavyaktAtmako viShNuranAdiH prabhurIshvaraH*

*shuddhasatvo jagatsvAmI nirguNastriguNAnvitaH ||1.9||*

*Translates to*

*The nature of the single unmanifested is equilibrium of the infinite potential capabilities, the pure true essence of the master of movable/unmovable is linking the unqualified with the directional quality.*

When an indeterminate in equilibrium is disturbed, another equilibrium tries to get established. The indeterminate trying to re-establish equilibrium forms the movable/immovable, which is changing of the continuous bits formed the single continuous Brahman to form changing activated or dormant data. Thus, the indeterminate Brahman takes on a quality of "some direction" that leads to equilibrium. We need to realize that direction need not always have the meaning of a "direction in space". Here we do not have space, hence the meaning of direction is anything other than the indeterminate to associate an order that goes towards establishing equilibrium. The Nasadiya Suktam also says this, as I had said in the section of "The dormant state of data":

*tama AsIt tamasA gULham agrE aprakEtaṃ salilaM sarvam A idam |*

*tuchchy EnA abhva apihitaM yadA AsIt tapas as tan mahinA jAyat aikam || 3 ||*

*Translates to*

*Unbound, inert, foremost covered by dormant, all is indistinct, unsteady is this, minute in that enormous enveloped, if unbound heat*

*exists, there in that mighty is produced that one foremost agitation.*

Thus, the introduction of an agitation in the Brahman causes it to re-establish equilibrium which forms the inner workings of reality. The questions this raises are "what is equilibrium w.r.t to the continuous whole?"

To understand equilibrium, we need to follow through with the thought of expansion of the movable/immovable into the gaps? If what is said in the BHPS is true, as I have written previously:

*sUrya natvA grahapatiM jagadutpattikAraNam*

*vakShyAmi vedanayanaM yathA brahmamukhAcCautam ||1.6||*

*shAntAya gurubhaktAya sarvadA satyavAdine*

*AstikAya pradatavyaM tataH shreyo hayavApsyati ||1.7||*

*na deyaM parashiShyAya nAstikAya shaThaya vA*

*datte pratidinaM dukhaM jAyate nAtra saMshayaH ||1.8||*

*Translates to*

*The cause for the movable/unmovable is nothing else other than appearance of the acquired pattern, the rendered guided by knowledge of the closely woven direction of creation. The regular distribution of density across the every rendered truth and continuous aggregation of the given energy, facilitates binding of scattered movement. It is not given because of remote guidance nor aggregation of that movement, there is no doubt that, it is produced by the contraction of the gaps between regular divisions.*

The regular divisions of activated data or continuous bits expand into gaps contracting the gaps and continuously aggregating and this causes the movable/immovable, then, that is the process that is happening in the Brahman to establish "equilibrium". So, when that expansion stops, equilibrium should have been established. So, the question is when does the expansion stop. The BHPS continues to tell us this:

*saMsArakArakaH shrImAnnimittAtmA pratApavAn*

*ekAMshena jagatsarva sRjatyavati lIlayA ||1.10||*

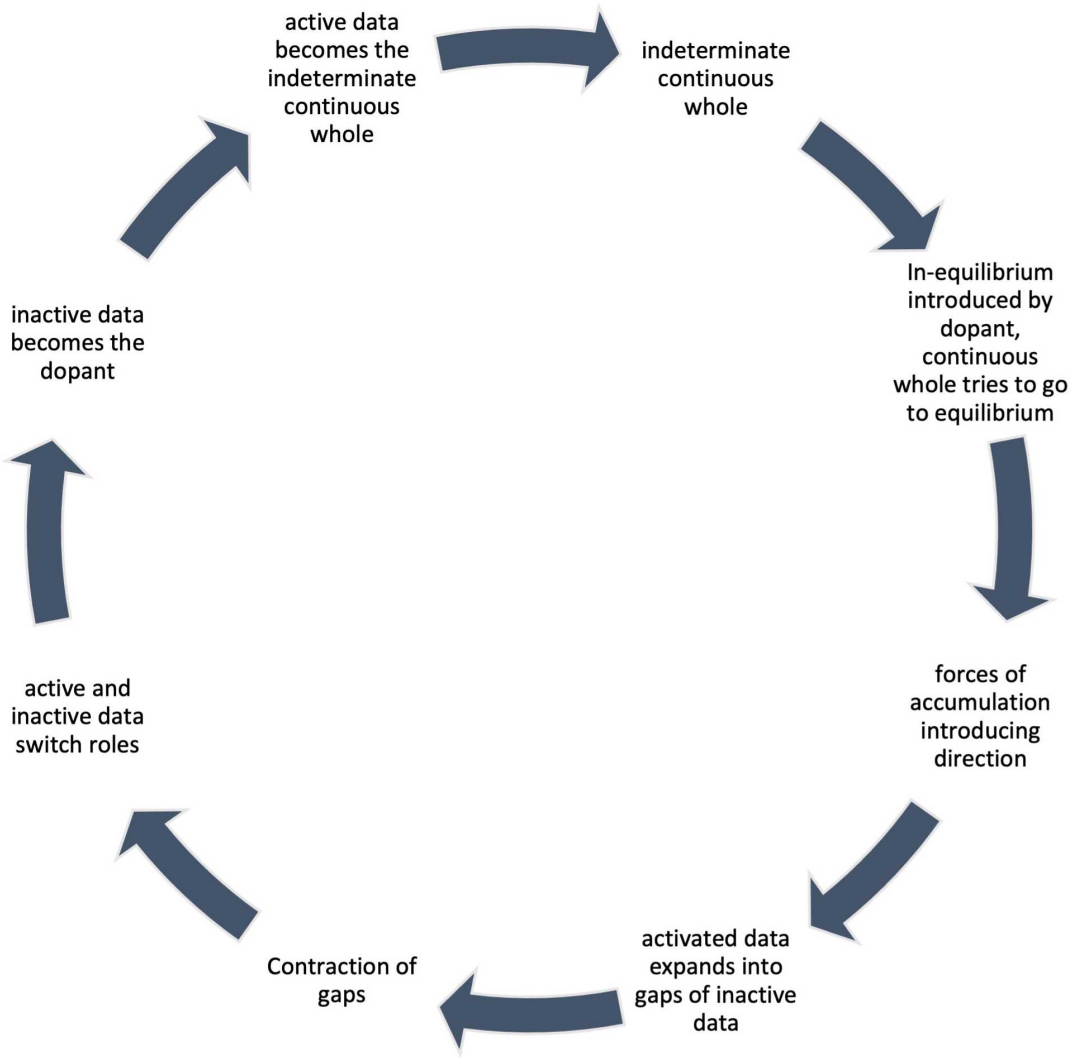
*Translates to*

*The work intent of the cycle of existence, is the nature built by mixing thoughts, before acquiring thermal energy, in that single part, which is the entire movable/unmovable created, until it attains appearance.*

If we read the above verse, it tells us that it continues to go through the “cycle of existence” till it gets an appearance and this is the first equilibrium. If it keeps expanding further even after the appearance, the gaps should have completely contracted to have no gaps at all and at that point, all expansions and movement should have stopped.

This means, the whole continuous switched from a mix of indeterminate and activated, inactivated data to a continuous whole indeterminate again. Now, that we have the extra information of, from which the indeterminate was created, we can deduce that the dormant or inactivated will become an agitation in such an indeterminate, because the continuous whole is an indeterminate created from activated data. Thus, in this state, if a continuous bit of original dormant data exists, then, that becomes the dopant “active data” in this indeterminate and that which “was active data” switches to become “dormant data”. Thus switching between active and inactive. This cycle should now continue till the current active and inactive switch.

This is the first, topmost level cycle that is established in the continuous whole. The cycle that keeps switching between active and inactive states of the continuous whole in the Brahman. Obviously, now that we have reasoned to this, we can see that the current active will do the same till the current active will become inactive and hence, the cycle continues. The below shows the cycle:


active data becomes the indeterminate continuous whole
indeterminate continuous whole
In-equilibrium introduced by dopant, continuous whole tries to go to equilibrium
forces of accumulation introducing direction
activated data expands into gaps of inactive data
Contraction of gaps
active and inactive data switch roles
inactive data becomes the dopant

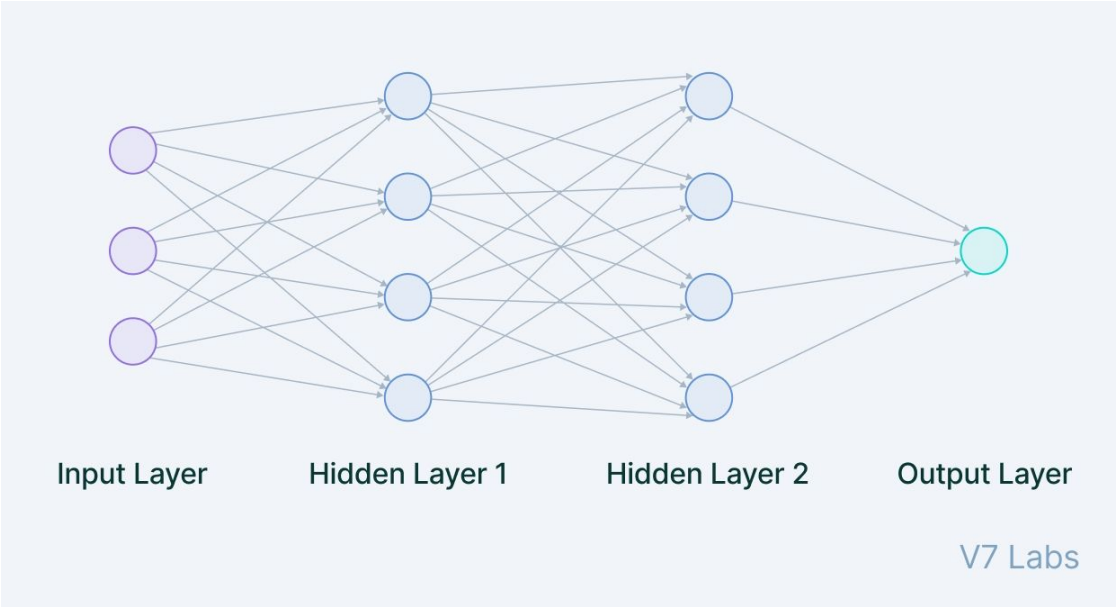
So, how can these cycles be controlled? These cycles can be controlled by controlling the expansion of the currently active data into gaps, by changing various parameters such as, the speed of expansion, the direction of expansion and so on. This raises the question of "How are these parameters controlled?" This is explained in the above verse. According to this verse, it is "work intent of the cycle" that controls the cycle.

## Learning methodology

Now, that a change is started off with a dopant, how does the guiding principles act to form a learning algorithm? What is the learning methodology that is used to create knowledge?

In a neural networks, the learning is in the form of multiple neurons coming together into various layers from the input layer to the output layer with one or more hidden layers to produce an output

for an input given. Based on the network form, these nodes can feedforward or have a backpropagation, but the end goal is to create the network. Each node learns the weights, bias, activation functions and so on. As shown below:




Neural networks apply multiple types of learning algorithms based on what is required to learn and create a node with the appropriate weights, bias and functions. Some of the learning algorithms are linear regression, logistic regression, naïve bayes, k-nearest neighbours, k-means, Markov decision process and so on many algorithms. Each of them are used based on the use case and type of learning wanted, supervised, unsupervised or reinforced learning. Many of these algorithms are used to cluster data, classify the data, determine weights, bias and other parameters. There is no one single algorithm that will run throughout the learning process repeatedly on variety of data and work to get the required learning.

Further, the neural networks learn to be a single linear flow logic (similar to writing a multiple if-else clause logic), which, once the learning has completed, runs the logic for any given input data to give an output based on the learnt logic applied to the input data. It is a “single focused intelligence network” or a “single machine learnt logic” as learning is focused on only one narrow part of some bigger domain. For example, neural networks are used to “learn and suggest semantic changes to a sentence in the English language”. This is the goal of recurrent neural network. Another example is learn to identify features within an image. This is a convolutional

neural network and so on. As can be readily seen, these are focused on very narrow intelligence in a single domain.

What also should be realized is that for neural networks, knowledge and intelligence are one and the same. In fact there is no conscious recognition of the fact that there is knowledge that is present in data and there is intelligence that is applied over this knowledge to get some conclusions. There are inherent processing that accumulates data to learn the intelligence as weights, bias or activations functions. But they are part and parcel of a process that learns and not stored specifically for later use. Clustering is one such sub-step that can provide knowledge contained in data, but is just used as a means of grouping data to learn from rather than anything else. Many clustering algorithms are present. Again all the algorithms are based on mathematical representation of data that can compute distances between data using various means. The k-means clustering, mean-shift clustering and many such clustering methods.

Unlike the neural networks, inner workings of reality is a self-sustaining continuously changing, churning whole, trying to re-establish an equilibrium and thus activating more and more changes. It is a continuously churning data that continuously forms knowledge and hence has a continuously changing logic. Its goal is not knowledge or intelligence. As I have explained in the section of "Fundamental architectural component", we are not creating a logic network here, but a data network, in which the stored data representation directly has the knowledge in the form of data, which is inherently the learning. This makes this system cover a very large area of knowledge. Knowledge is only limited by the kind of data available and the knowledge inherently contained in that data. If data is available in multiple domains, then all the "domain knowledge" is contained in a single continuous whole.

The beauty of this type of knowledge is that, since the *original data is just appropriately observed to form knowledge,* rather than a modified version of the observed knowledge being stored and used, *if the data changes, the observation changes* and hence *appropriately the logic changes without any delay* or extra processing needed. The data change is the only required event and all related logic also changes instantly. Also, unlike the neural networks, we are not working with multiple types of algorithms to achieve different different purposes. The same foundation method i.e., integral of differentials got by observing data overlayed over itself, is applied at various levels on various data automatically and gives us knowledge associated with the data. A single foundation

method that behaves differently based on data and hence gives different learnings.

When we look at the way the internals of reality works, we find that the coming together of the cycles at the top-most level can be seen as an environment that has the field of vision as all the activated data. From this top-most level children cycles can be formed with reduced field of vision in a hierarchy. Given that the learning is represented by knowledge and is stored directly in data, as the data is rolled up from the child levels to the top-most level, the knowledge also can be rolled up. Further, from the top-level data knowledge can be decomposed down into modular child data which contains modular knowledge within itself. The learning thus is present all the way from all the lower-most child-cycles all the way to the top-level cycle, each of them following the same compose-decompose philosophy infinitely.

It should be recognized that in such an environment, there is not just one line of logic is learnt, but "multiple lineations of logic" learnt. The number of lineations of logic built are based on a number of factors. Some of the factors are the different ways in which the data can be grouped to build logic, the different observers that can be self-created from the data and many such factos. So, how is the knowledge built? Here is where the continuous real-time foundation algorithm is used. As I have said, the basis of all internal working is this foundation algorithm and is used over and over again in each and every step to create knowledge, sub-knowledge and so on. In this case, as data is streamed, the divisions overlay over each other. The repetition ensures the observability of a repeating pattern in the "*form of persistence*", as one division overlays the other. This observed repeating pattern is knowledge. When multiple of these type of overlays form multiple repeating patterns, multiple knowledge is formed. This is the first step of the learning algorithm. It should also be realized that the observer has the capability to observe multiple repeating patterns within the same location of various different properties and values.

An example of this can be seen in the rut that is formed on roads as the vehicles pass over the same path over and over again, as shown below:

As can be readily seen in the above picture there is a lot of knowledge available to be read here. For example, the number of vehicles have travelled in crisscross directions on this road, with a tyre treads as is seen. It has rained both before and after the vehicles have passed and so on. The example that I have given are just descriptions of data that is already present in the picture. We can either describe the knowledge or leave it as data as I,s present as a picture. To be clear, it needs to be pointed out and understood that, the picture above is also just a “description” of the same data. One describes the conclusions in words, the picture is data described in light reflections on the road and images with conclusions from this, formed within our brains.

The actual data is that activated data which renders as the actual mud-road on which when the data that represents light is reflected, my eyes receives it and within my brain has formed this image. It is important to see this sequence, even if this is not the complete sequence of formation. It is equally important to understand which data contains the actual knowledge contained in it. What we look at as knowledge or intelligence, is that description that is formed within our brains and we are able to articulate it as words or pictures. But, this is not truly where the actual knowledge is contained. It is because we try to emulate *this* formulation of the description of knowledge as learning, we hit problems with the AI that we create.

It should be understood that the learning of the knowledge started at that activated data that first rendered as that mud road in its pristine condition when no vehicles had passed through it. Subsequently, as the vehicles travelled on it, that “activated data” got overlayed over and over again by the “activated data” that render as

the vehicles. This can be seen as a division of data containing the data of the vehicle, streaming on the division containing the “activated data” of the pristine road, causing the foundation method to execute which accumulates the differentials. The result of such an execution of the algorithm is that data which contains the knowledge of the vehicles that have passed over it being collected over the data of the mud road. Thus, the learning cycle is the repeated execution of the foundation method on various continuous bits of data. Once, the knowledge is extracted, description and intelligence are applied on that knowledge by using the architectural component cycles.

The above method described is also a clustering methodology which automatically shows the data distances and clusters related data. Again to bring out the reason why this algorithm works is because of the appropriate data representation that automatically realizes an algorithm. By effectively reducing the number of steps required to compute differentials and integrals and the ease of doing it across all data has removed the time constraint that is present to compute it in discrete data. So, if, like a ribbon moving on a spool in a movie, if the data is rolled over each other and continuously observed together, automatically similar data is grouped together from that data set. But, what is more is that, here it is not viewed as a clustering method, because the goal is not to create groups. Grouping is the method, the goal is to infer knowledge from data, which is what is got when this is done.

This method cannot be applied effectively to discrete data representation, because for one data cannot be folded and observed to give the correct delta data, for two, a single bit only represents 1 or 0 and not the actual data. The fact that we need to use complex algorithms is because of the discrete-ness and complex-ness in our data representation.

## Learning with work intent

This accumulation of multiple changes as I have described above to form a continuously changing “activated data division” leads to “work intent” or in Sanskrit “kAraka”. It starts out as a continuous minute bit of activated data set, over which as multiple activated data sets accumulates, it takes on a “retained form” because of overlay of the occurrence of the same change over and over again at the same or displaced location. According to the BHPS, this takes on the shape of “miNa rAshi” or the “multitude with comet shape”

which is intent. The formation of the work intent described as below in BHPS, as I have indicated before:

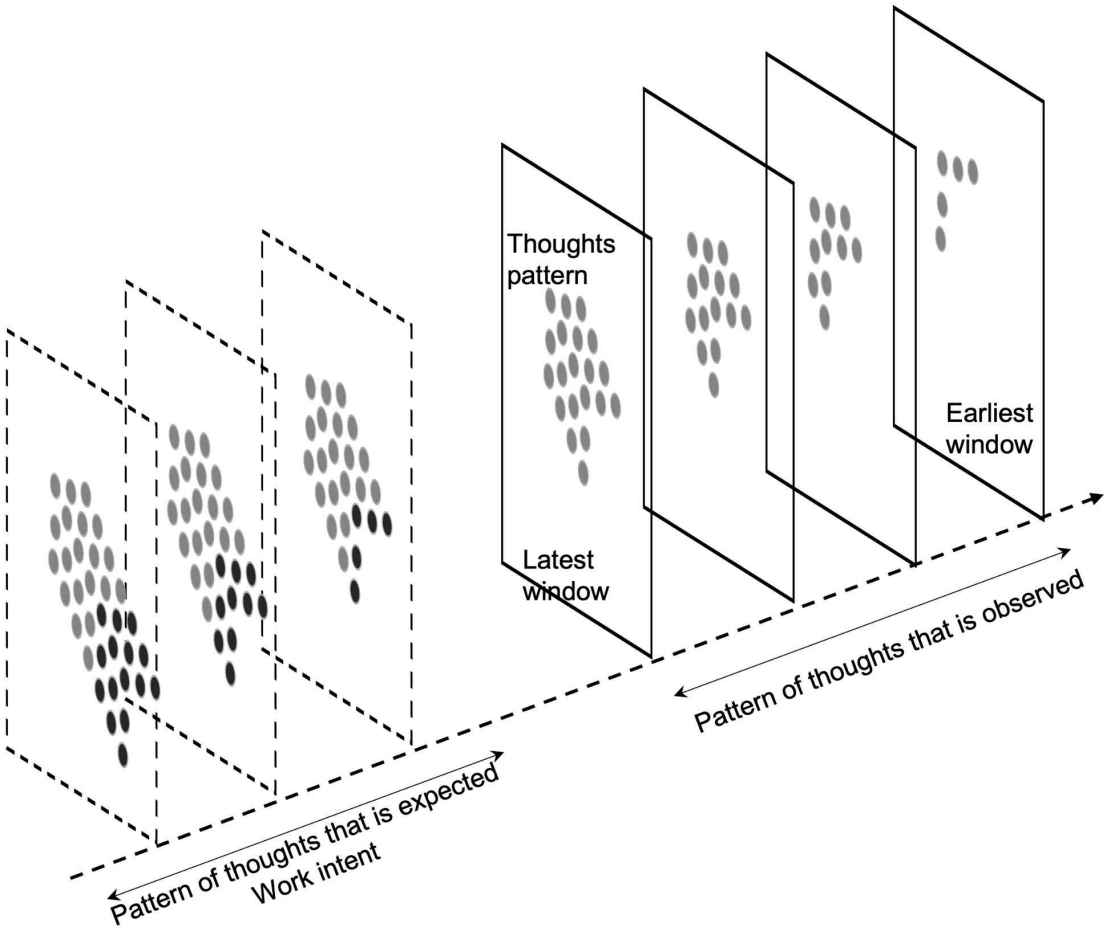
*saMsArakArakaH shrImAnnimittAtmA pratApavAn*

*ekAMshena jagatsarva sRjatyavati lIlayA ||1.10||*

*Translates to*

*The work intent of the cycle of existence, is the nature built by mixing thoughts, before acquiring thermal energy, in that single part, which is the entire movable/unmovable created, until it attains appearance.*

First we need to understand what "work intent" is. To understand this, we need to look at "what energy is defined as". As I have described in the book Surya Siddanta: Emergence of empirical reality, energy is the potential to change of the Brahman as it is traversed from one location to another. When a specific pattern is associated with this potential to change, i.e., the potential to change in a certain pattern, it becomes a specific type of energy. So, the potential for a specific regular oscillating pattern to occur is thermal energy. *The "work intent" is then the potential to be able to activate enough continuous bits of data to form that energy pattern*. As is said above, the way this work intent is built is by mixing thoughts that forms the pattern of thermal energy. We are a thermal energy system, hence we always activate various variations of the thermal energy pattern. Why thoughts? Because thought is defined as a specific base pattern from which every other data is formed as I have described in section on "the active state of data". The below diagram shows this:




In the picture we see that there is a pattern of thoughts getting formed. The first pattern formed and self recognized is at the right most window, which is the earliest window. On this as changes keep getting overlayed, the recognizable pattern keeps getting accumulated in the subsequent overlay, till we have the fourth window from the right. Till here the pattern has been recognized and observed, so that an appearance has already been made. Subsequently, we find that the "work intent" starts getting formed with the "potential to form the same pattern" again. This can be seen as a projection of the movement of the pattern or seeking of the same pattern again.

*I have represented the pattern as an understandable space dimension pattern and kept the pattern very simple. But, this need not the case in reality, the pattern is a thermal energy pattern. To know such a pattern, we just need to boil water in a large vessel (like the ones used to cook food for a large event) till steam rises and the heat spreads out from the vessel. Then subsequently, walk towards and away from that vessel, we can feel the pattern. It should also be noted, that I have created it as snapshots at regular discrete intervals as opposed to a continuous pattern that is typically present.*

*We need to see it as a continuous gradual change as we see a gradience in color.*

As we see from the above verse, the work intent needs to be present till an appearance is formed for that "work intent" which is the repeating pattern. For example in the above picture, I have created a right angle pattern of 5 dots and has recurred for 5 continuous windows before the intent is formed. If this repeating pattern cannot be formed, then it is not considered a "work intent". This is similar to driving on a road with a rut that has formed. If you have driven a car on such a road, you will know, once a rut has been formed, it is very difficult to drive a car away from the rut, because the wheels keep slipping into the rut. Thus, the basic requirement to a learning to become concrete is the recurrence or accumulation of a certain pattern some X number of times, hence forth from where it takes root as intent and progresses based on the logic learnt from that accumulation. I do not believe we have any such concept in our learning algorithms, because this is a requirement of a real-time algorithm. Since, our algorithms are learnt offline, this concept does not apply, because all data used in offline learning is used to compute weights or bias and other such values.

What is important to note is that in a single environment, we can have many such "work intents" formed. Hence a whole lot of "knowledge" is created in parallel which can be used in a number of ways to create intelligence for many domains, across domains and many such combinations simultaneously.

The next to be noted is that the computation of differential for the stream of data is computed as a change against each and every other activated data, since the data representation is continuous. There are no outliers. Outliers are only w.r.t a group of data. Because of the algorithm, automatically only relevant data will accumulate into a specific group. Again, since knowledge is represented directly as data, learning is just accumulating of the appropriate data according to the pattern they form, which is what "work intent" allows us to do. But, what work intent also allows us to do is control that progress of accumulation.

## The work intent cycle

In the AI algorithms we write, we typically stop at the learning methodology i.e., the learning algorithm stops once the network has learnt the logic from a training data. This is because the goal of a neural network or the AI we create, is to learn the logic contained in

the data. Hence, it stops here. Subsequently, it just reuses the logic built for various pieces of data. But, in the internals of the working of reality the algorithm does not end with just learning. The goal of reality is to achieve an equilibrium. Thus this needs to induce a learning that can go all the way to the goal of achieving equilibrium rather than stop at learning. The learning methodology is just a means to get to that equilibrium. Hence, there is a continuous cycle that flows till some specified equilibrium is reached.

It is important to note that the cycle that is followed to achieve that equilibrium is just an understanding and a description. Any changes to that description by itself or a wrong understanding of what actually exists, does not in any way affect the cycle followed to achieve the equilibrium. For example, the way an alternating current (AC) is described typically is: the voltage shifts polarity periodically and when it shifts polarity, the electrons flow in the opposite direction. But, this does not really mean that, in actual, the electrons are actually flowing back and forth, it can be looked at also as, that they are just sitting in a single place and wiggling. But, whatever is the actual, the description is so, such that we can understand. Changing the description to flowing of electrons does not make the electrons really flow.

Similarly, we need to remember that we could describe the establishing of the equilibrium as a cycle or just an establishing as an equilibrium, what happens is what happens. The process is described as a cycle in the Sanskrit literature because the ancient people seem to have recognized the cycle.

I have named this cycle as the “work intent cycle”, because the “work intent” is the one that drives the whole cycle and controls the learning methodology to continuously change. The start of the cycle is when a dopant or recognizable change is present in the indeterminate whole. The question is “who recognizes the change”? Here is where the Surya Siddanta gives the answer:

*bhAnAmaShTAkShivasvadritridvidvayaShTasharendavaH*

*bhodayA bhagaNaiH svaiH svairUnAH svasvodayA yuge ||34||*

*Translates to*

*Emergence of the 8 (curves), produced focal points, unrestrained three (gaps), 2 (focal points) is double thread configuration when transformed, observing this(pattern) within the series of becomes by itself without independence gives rise to a connected self*

As I have described in the book, “Surya Siddanta: Emergence of the empirical reality”, the appearance of a specific pattern of variation in the indeterminate whole is thermal energy. When that specific thermal energy pattern occurs and that pattern further continues to form a “double thread” pattern, as described in this verse, then, this pattern is able to recognize itself in the “indeterminate whole”, becoming an observer of itself. This is the most important point which breaks the need for an external observer and starts off the cause/effect concept. Many Sanskrit literature have referred to this as, “the self observing itself”. The Bhagavad Gita refers to this as “saMstabhyAtmAnam AtmanA” = “saMstabhya + atmanam + AtmanA = strengthened + self + by the self”. The BHPS introduces this concept as “protected by observation” or “adhipA” which strengthens and retains the self. In general, the same concept is described in different ways. The concept itself remains the same and can be described also as “the self gets the capability to observe itself” and hence “becomes the starting recognizable change”. This is where the whole cycle starts.

Again, we need to understand that we tend to associate words with certain meanings. When we talk about “self gets the capability to observe itself”, it does not mean that it observes and forms opinions, judgements or any such extra emotions or thoughts that are formed by us when we observe ourselves or others. “Self observes itself” can just be seen as that state where the formed pattern does not scatter away and stays stably and is able to accumulate more. To understand this we can draw comparisons with a sand-castle. When starting to build the sand-castle, if we are able to pick up the right place where the waves do not wash it away, has the right amount of moisture to stick together and form the foundation to form the support, over which a bigger structure can be built, then, this foundation can be thought of as that “self” over which more can get accumulated.

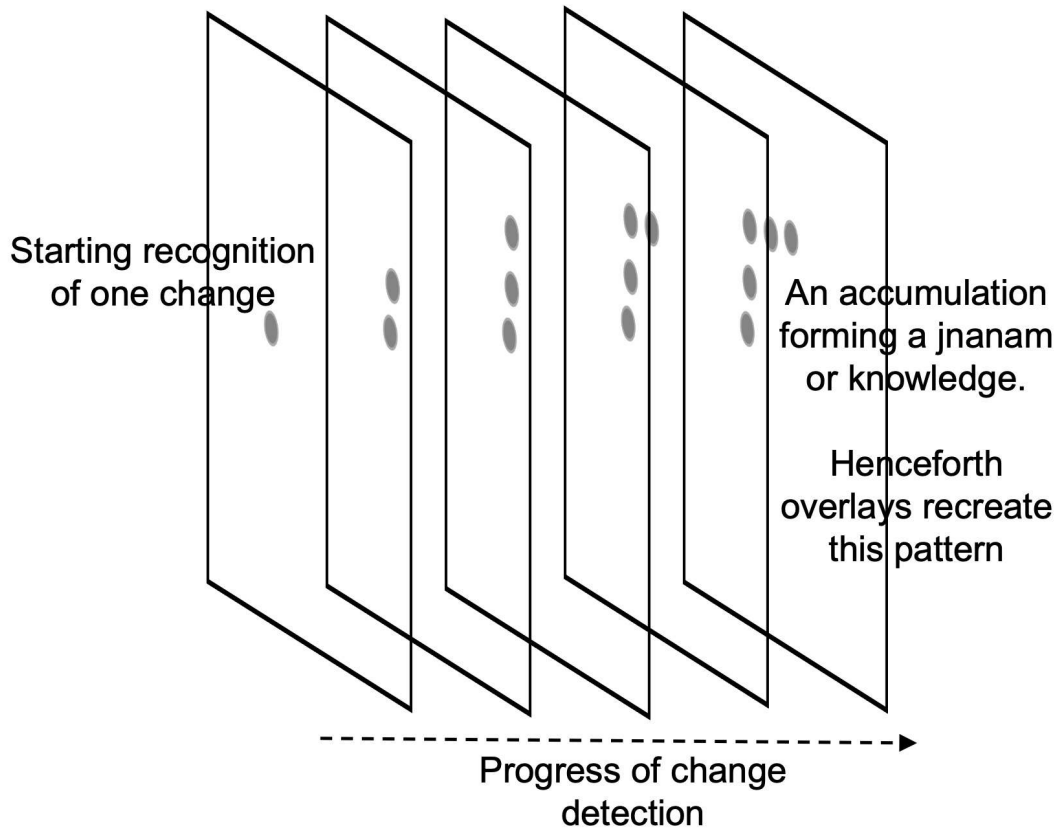
We typically tend to understand concepts assuming an isolated environment because we are trained to the notion of experimenting in a controlled environment. But, that is not the case here. In here, we are studying and understanding reality as is without making changes to it, where changes occurs spontaneously. Thus, while the description of the concept that starts off the starting point can seem isolated, it should be recognized that this could be occurring in parallel across the indeterminate whole in multiple locations, can be occurring when there are already established cycles present and many such combinations. If this had occurred when there was

nothing present at all, and it was a pristine indeterminate Brahman, then it starts off the whole reality, but when it occurs after reality has started, then it is just a creation of a thought series or an individual and so on it goes. The same concept can be applied recursively for everything, same as the compose-decompose philosophy that I have explained before.

**Primary energy with capability to recognize change (Mahadeva and Shakti)**

When we look at the creation of the starting recognizable change, we find that a few steps of the cycle are already present as a part of it. The first is there is a potential for a "series of changes" to be present in the Brahman, only then can a pattern of change be recognized by itself. This potential to change to form the primary pattern is called the "Mahadeva" or "prime energy". This "prime energy" has acquired the capability to "recognize the change and observe it by itself" and hence started it. This capability is Shakti in Sanskrit. So, we have a "Mahadeva with the Shakti to recognize change and observe changes within itself".

Over this recognized change or dopant, the learning methodology establishes a "work intent" by using the foundation algorithm. This can be seen as the "accumulation of the observations of changes made by the Shakti". As one division overlays the other to form the work intent, the observations are accumulating in that single window of observability. This observed accumulation now becomes "jnanam" or "knowledge". Given we are looking at an environment with many such "observations accumulating", to each "jnanam" there is a "para jnanam" which is "remote knowledge", i.e., remote to the "change observed". That is, to each instance of Shakti that observes the "pattern of self", there are other remote "patterns that have recognized themselves as a self". Thus, this is the next step in the cycle, the creation of the "jnanam and para jnanam". The below a representation in space dimension of how accumulation causes recognition of jnanam or knowledge:


Starting recognition of one change
An accumulation forming a jnanam or knowledge.
Henceforth overlays recreate this pattern
Progress of change detection

To be noted as I have said all through, the diagrams I have drawn is just a depiction of what is happening as a representation in the space dimension. It should not be mistaken for this to be what is present. All of circles and windows are just representations of data which can mean anything, based on the observer's interpretation.

Again, all established words in english seems to have a preconceived notion that is evoked in the brain. So, when using these words, there needs to be an explanation to clarify the correct meaning. While, I have used the word knowledge here, knowledge is again a very subjective word. So, to clarify, given we are looking at a data-based algorithm and data-based meanings, "knowledge" is also based on the change accumulated. Thus, when changes accumulate, we can see them for example as a "series with a linear rate of change" or a "series of exponential rate of change" and so on. This becomes "knowledge of the type of change" that has occurred.

Another pre-conceived notion that we need to break is in terms of "Shakti" or the "capability to detect change". When we write algorithms in our computers, we write it as a processing algorithm. So, for us the processing is progressing and the data is getting modified based on the processing done to the data. So, the data stands still while the processing is moving. But, this need not be the case. The inner workings of reality being a data-based system, the

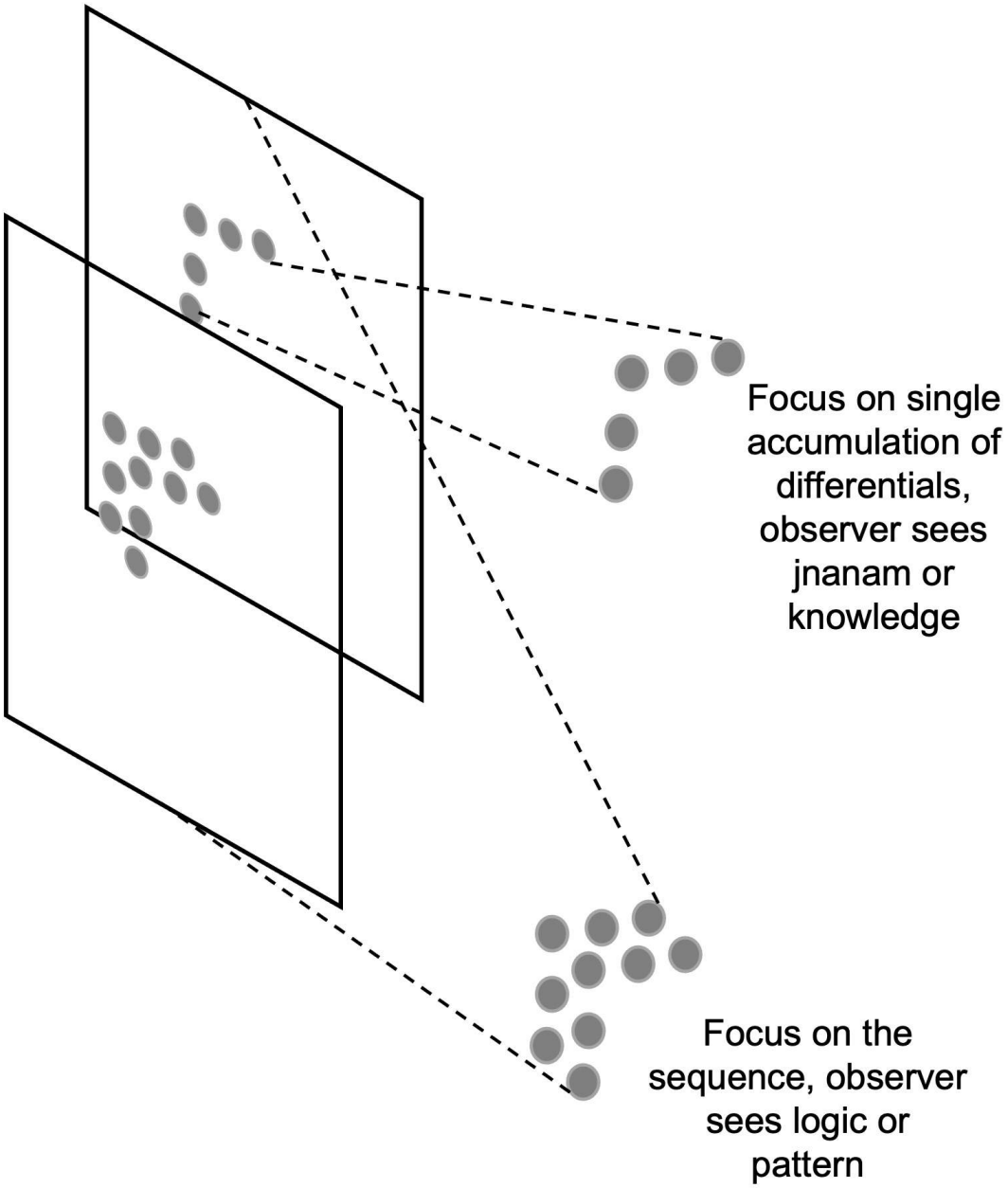
data is being streamed, i.e., the Brahman is churning and at a specific region the churn is observed to detect change by overlaying one change over the other. Which implies the data is moving, the processing or observer or Shakti or capability is at stand still at a single region. To understand this, we can think of a storm and ourselves w.r.t it. There we are at stand still with the storm passing around us, we observe the storm, we pass from the edge of the storm to the eye of the storm to the passing edge of the storm and see and feel all the ravages of the storm. Once the storm has passed over, we are left with all the impact of the storm as an experience that is left over. This can be seen similar to that.

**Logic to form a creator (Saraswati and Brahma)**

The next step in the cycle is to accumulate the knowledge and create the logic. As many jnanam and para jnanam accumulate, a logic is created called “Saraswati” in “Sanskrit”. While “Saraswati” is translated as “goddess of knowledge”, I believe it is “logic derived co-relating individual knowledge”. Again, while I have called it “logic”, it should be understood that this is not the “complex logic” that we write in computers with lots of if-else, while loops, objects and so on. We need to keep reminding ourselves that we are dealing with purely data-based system where data represents the meaning. Here when we talk logic, we are purely taking “progression of data” in a certain format. It should be recognized that *even establishing what is a sequential progression of data becomes a logic*, because this also is not established as of yet. For example, if we had a knowledge series created as: “series with a linear rate of increase of change” followed by a progression to “a series of steady no rate of change” followed by a progression to “a series of linear rate of decrease of change”, this series of knowledge becomes representative of “logic” or “Saraswati”. As can be seen, I have given examples that seem “very regular in nature” for easy understanding, but this need not be the case and mostly it is not the case at all, because nature, as we all know, is very haphazard to our understanding.

So, how does this logic get created? Again the foundation method comes into play here. If we have an observer that can observe changes as one division is overlayed over the other, the observer can observe just the jnanam when the focus is narrow and limited to the accumulation of individual changes. When the focus is expanded to a group multiple changes that form many jnanam then the observer observes the pattern of the jnanam or the series of

knowledge that is retained as the data changes. This is shown in the below diagram as a representation in the space dimension:


Focus on single accumulation of differentials, observer sees jnanam or knowledge
Focus on the sequence, observer sees logic or pattern

Once, “logic” is established, the next step in the cycle is the creation of “Brahma” or the commonly translated term, “creator”. But, I believe, that Brahma is the “creation pattern” commonly used in java applications called “factory of factories pattern”. Hence Brahma can truly be compared to “a factory of factory of resources” that can create more factories that create resources. So, what is the resource that Brahma creating and based on what is it being created? It is important to understand this to understand how Brahma creates the resource.

**Resources to establish equilibrium (Lakshmi and Vishnu)**

Lakshmi in the normal sense is interpreted as “money” in our world. Sometimes it is abstracted to “currency”, meaning an external exchange that applies value or worth of a person, article or action. But, if we look at it from the actual context of the “cycle of existence”, Lakshmi, in my view should be “resources”. So, what are resources according to ancient literature? As I have said before, words invoke certain meanings in our minds. We need to break all these pre-conceived notions to understand the inner workings of reality. Resource for us translates to something useful external to us which we can use to create something else. We should not forget, what we observe in the observable world around us is just that, an appearance. A transformation of underlying something to this appearance. We are trying to understand that underlying something is. So, the question we have to ask ourselves is “what will appear to us as a resource”? It should also be remembered that the underlying something is purely data interpreted as patterns within patterns as I have been explaining, it can be called Saraswati or jnanam, but they are all patterns recognized in the data expanse. With this, the BHPS says this about resource:

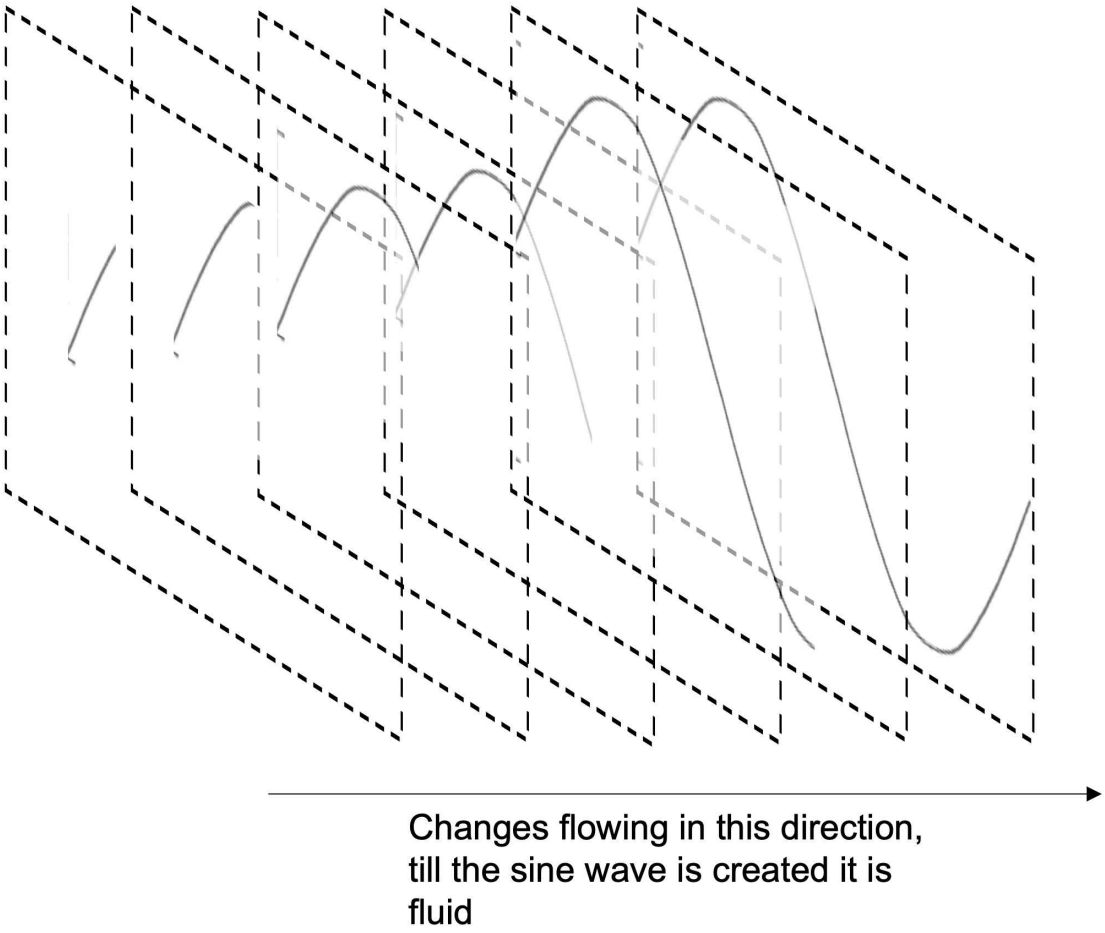
*vahniH samIrashakrau ca dhanado jaladastathA*

*viShameShu kramAjjneyAH samarAshau viparyayAt ||6.28||*

*Translates to*

*The strength of fluidity of the agents that give resources exhausts the fluidity thus the uneven seeking series impelled by knowledge clustering at the dopant inverts*

Given everything ultimately is just a pattern, the first to understand what is “flowing and fluidity” w.r.t patterns. For a liquid, we all know what fluid and flowing means. Flow is the rate of change as the molecules are passing through a cross-section of finite space and fluidity is the ability to flow or changeability of that liquid. Similar definitions can be applied to patterns. To explain, let’s look at a “sine wave pattern” and traverse it from the start till the wavelength, as changes accumulate to form the sine wave. It can be depicted as the below frames:




It can be seen that till the wavelength of the sine wave change is reached, the change will flow and be fluid in nature. This is because the sine wave changes at every point, as shown above and change is evident to the observer as it changes. Once the full sine wave is formed, the fluidity will stop because sine wave is a repetitive pattern. Hence will hold the pattern at the last frame shown here as the divisions are overlayed one over the other, even if progress is partial at a time. It will look as if some one is tracing the sine wave over an existing sine wave. This is what the verse is saying. When the fluidity stops that final something over which it is over and over again traced, gives us a resource. If the same pattern keeps repeating then, this will not be phased out and we get a resource of a sine wave.

At this point it hardens into Lakshmi or resource. So, Lakshmi is a hardened pattern that holds steadily. It can appear as anything to us, not just money or currency or value. In my view, in our world, any matter is Lakshmi, because it is a steady pattern that is held in a steady hardened fashion. Again, to be noted, what I have represented here is just a depiction in space. A change of some property value can be in the form of sine wave, but it should be noted that data is of

any property or value that is observed and take on any form. Now, going back to what Brahma is, Brahma is the ability to create these hardened patterns to form Lakshmi. What is needed for this is the correct set of work intents that needs to be held steadily, so that the pattern is keeps occurring over and over again to harden the pattern. In Sanskrit, this hardening is called “krUra”. The BHPS says this about this hardening:

*utpAtakAlasaumyAkhyAH komalaH shItalAbhidhaH*

*karAladaMShTracandrAsyau pravINaH kAlapAvakaH ||6.38||*

*daNDabhRnnirmalaH saumyaH krUraoatishItalaloamRtaH*

*payodhibhramaNAkhyau ca candrarekhA tvayugmapAH ||6.39||*

*Translates to*

*By the increasing appearance of series of changes having pleasing indistinct transformations in the surrounding cold, opens wide the transformations exists of the sharp edges, familiarity purifying the series of changes. The pure nourished force having transformations, hardening in the cold obtaining imperishability in the indistinct unsteady fluid container and (establishing) a line of transformation protected from the connected others.*

A few things to note here is the use of the term “in the cold surrounding” or “shItal abhidhaH” in Sanskrit. This is used throughout the Sanskrit literature. As I have said before, we are a thermal energy system, hence our primary pattern is thermal energy pattern. Any other pattern other than the thermal energy pattern becomes a “cold region” for us. Thus, every literature refers to a formation in the surrounding cold.

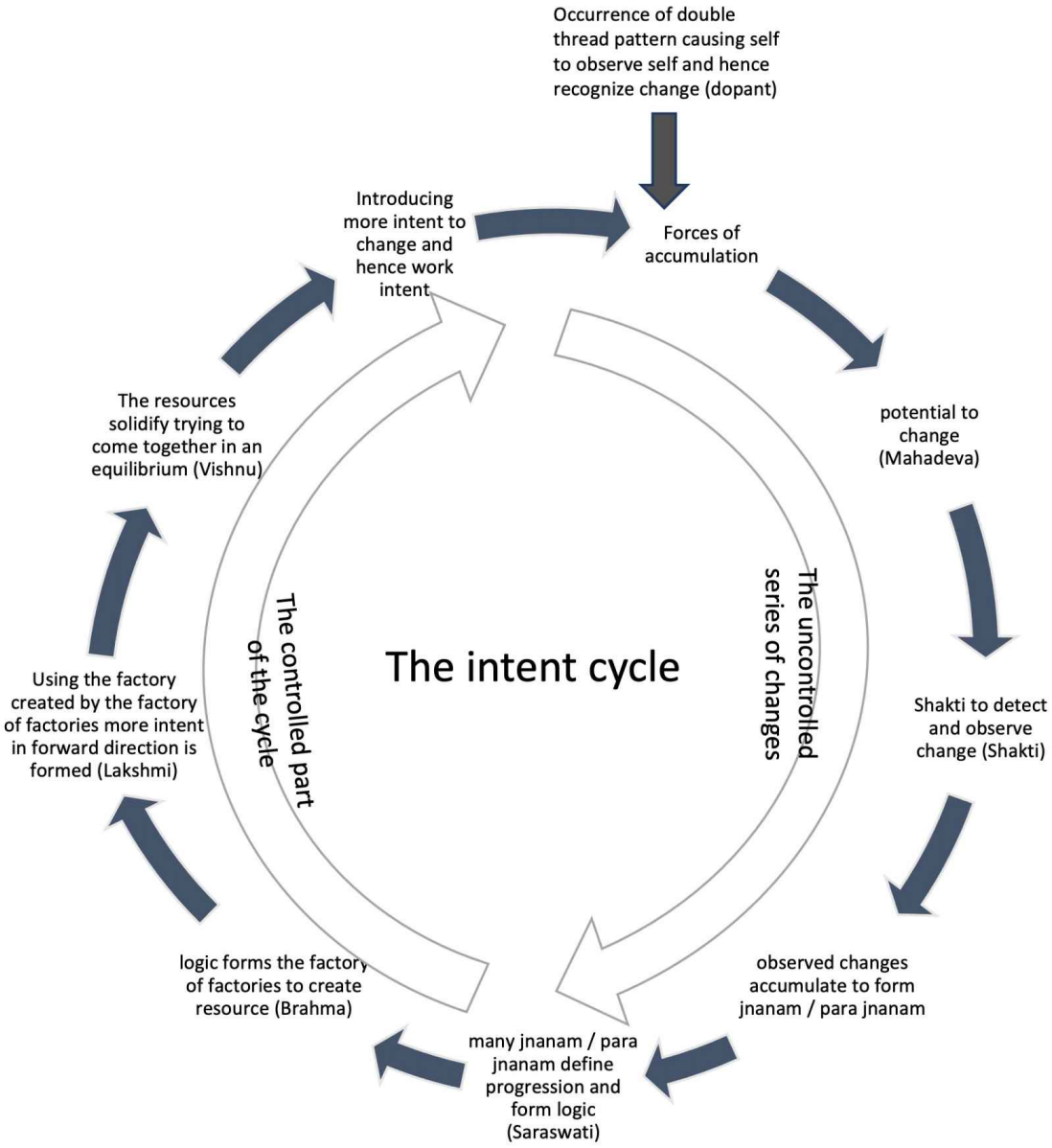
The next to note here is the transformation. As I have been explaining, we can focus and see just a single change or we can link together an accumulation and see a series which when transformed can give me a knowledge or accumulate even this and see the logic and so on it continues. So, from the above verse, if we can take a series of changes, observe it as a transformation by weeding out the outliers because of the foundation method which is equivalent to purifying the series of changes and keep nourishing it with the same series of changes, it should harden and become imperishable. This now is Laksmi or resource and be protected from changing.

It should be noted that I have depicted in the above an isolated pattern. But this is not the case typically in nature. Many patterns will be forming and being phased out without hardening. Thus,

many such resources that are formed of various different patterns cause interference with each other and try to come to an equilibrium. This is Vishnu, causing more work intents. More work intents because the expectation of a pattern changes to the pattern of interference. These work intents introduce forces of accumulation.

**Forces of accumulation**

As this is a cycle, we can start at any point and keep traversing till it comes back. I have started the explanation assuming a work intent is present and continued to explain the subsequent phases. I did not explain the forces of accumulation before, because this needed context of all the subsequent steps in the cycle. The below shows the whole work intent cycle I have explained above:


Occurrence of double thread pattern causing self to observe self and hence recognize change (dopant)
Forces of accumulation
potential to change (Mahadeva)
Shakti to detect and observe change (Shakti)
observed changes accumulate to form jnanam / para jnanam
many jnanam / para jnanam define progression and form logic (Saraswati)
logic forms the factory of factories to create resource (Brahma)
Using the factory created by the factory of factories more intent in forward direction is formed (Lakshmi)
The resources solidify trying to come together in an equilibrium (Vishnu)
Introducing more intent to change and hence work intent
The intent cycle
The controlled part of the cycle
The uncontrolled series of changes

As we see above, all of this starts off at some agitation introduced. A varying change is present in that continuous whole because of the continuous whole trying to attain an equilibrium induced by the first agitation in the continuous whole which was in equilibrium, this agitation becoming the first dopant. As the continuous whole churns and changes towards that goal of trying to attain an equilibrium, that goal is recursively traversed down to the smallest "continuous bit of activated data" that is formed because of the agitation. This introduces forces of accumulation. In BHPS this is called "collective pull" or "saMkarShaNa" that allow the continuous bits of data to re-arrange themselves in changing patterns. The BHPS says this about the collective pull:

*saMkarShaNashca pradyumnoaniruddha iti mUrtidhRk*

*tamaHshaktyAanvitA viShNurdevaH saMkarShaAbhidhaH*
*||1.15||*

*pradyumno rajasA shaktyAaniruddhaH sattvayA yutaH*

*mahAnsaMkarShaNAjjAtaH pradyumnAdyadahaMakRtiH*
*||1.16||*

*Translates to*

*By the collective pull of the penetration and the unobstructed appearance of interference, this form is rendered, its inert capability linked to the collective pull of the surrounding energy of the equilibrium. The appearance of interference by dopants to the unobstructed capabilities connected to that true nature, propels production of a great collective pull, from then the appearance of interference destroys the incomplete.*

As can be seen from the verse, it says that the collective pull of the penetration which is each individual intent is linked to the collective pull of the surrounding's equilibrium. And introducing a dopant propels the collective pull.

So, what are these "forces of accumulation" or "collective pull" or "saMkarshaNa". This can be best understood by looking experiments of oil and water. When oil is poured into water, we find that it accumulates on the top. Shown below:

Now, if salt is poured over this, the salt tends to sink, pulling the oil along with it. Salt tends to dissolve in water. When it does dissolve, it releases the oil it pulled down with it, which then rises again above to the top. Thus we have the dancing oil experiment. The result of this experiment can be analysed in a multiple ways. In-fact any scientific experiment can be analysed in multiple ways.

Scientific experiments conducted can be analysed to understand just the appearance in this reality as we typically do and as is done in the above experiment. Such analysis allows us to define the limits of this appearance and hence define the rules that govern the appearance. This is modern science. Rather than just looking at the appearance, if we understood that what is visible is just what it "*appears to me as this*", there is an underlying mechanism that is

driving its appearance as thus, then, we can use the same experiments to question and understand the underlying mechanism that is driving the appearance. This allows us to use the results of the same experiment to also understand the working all the way from the truth to the appearance. Such a process can be compared to the debugging of computer applications. When we are debugging an application and want to understand why the UI is behaving in certain way, we do not just debug the UI, but also the backend that is powering the UI. The first level support just look at the usage and debug and try to fix it, where as the third level support or the developer support involves going through the code and finding the problem and fixing it. To understand the inner workings of reality we need to be the developer support.

So, we need to question, what does this above experiment indicate to me. The first most important aspect to understand in this experiment is that, it is an experiment that does not have any external process involved that forces the result to be in a certain manner. All that occurs are just natural forces and processes. The only external action is the act of pouring in the oil or the salt into water. According to the standard scientific explanation, this is attributed to the fact that oil is less dense than water and hence tends to collect on top. It is explained to say that the molecules of water is heavier than the molecules of oil and hence they cannot mix. So, how can we analyse this experiment differently.

We start at the point that molecules are also appearances based on some underlying mechanism. We can keep splitting the molecules down to its smaller and smaller pieces as is done in modern science, but this will just keep taking us down to the smaller and smaller pieces of appearances. Now, instead of staying still in the appearance, let's say that we shift the focus to that which appears to me as thus i.e., matter. As I have been explaining before, if everything starts at an indeterminate expanse and starts with continuous bits becoming active data and a continuous series of these bits being folded over to give me a pattern, then this "molecule" that I refer to, also should just be one such pattern that hardened. Let us keep the way the pattern is formed for subsequent discussion, we are still only looking at the structure that is formed. We know that in some form it has to boil down to the pattern of continuous bits i.e., activated data.

When we talk about molecules of water being heavier than molecules of oil, we are infact referring to the number of electrons and protons that form the molecules. Hence, it should imply, "that

pattern that forms electrons or protons and so on smaller bits of the molecule in appearance" accumulates together to a molecule. In one molecule the number of such patterns that have accumulated is more than the other molecule and this difference appears as difference in density of patterns. Thus we can see this as two parallel running concepts, the pattern backend powering the appearance of a molecule.

Now, let us look at the fact that oil and water molecules do not mix. While, in appearance we can add a rule that says heavier molecules do not mix with lighter molecules, the same cannot apply as an explanation when we are looking at the patterns that form the internals of that molecules. Here, we are looking at continuous pieces. The question thus turns into "Why does one pattern not mix with another pattern?" This leads us to the "forces of accumulation". As we keep folding one continuous bit over the other, we find that based on the pattern, the accumulation of differentials tend to take on certain set patterns, automatically weeding out outliers, accumulating similar patterns together to form other patterns and certain patterns not having an effect on other patterns basically because the properties don't match. The observed resultant pattern again depends on the observer.

Thus, there exists this "force" that ensures that only some types of changes accumulate together because of some order established by the "observer", which inturn is driven by the goal which is to establish an equilibrium. This establishes an order (dharma in Sanskrit) by which patterns are created. This drives how the data accumulates with each other as more and more data is activated. This is called "forces of accumulation" and drives the varying changes to be present in the continuous whole and dictates how the Mahadeva is formed, and has a direct impact on the potential to change. Subsequently the cycle continues to the Shakti to detect change and continuously the cycle goes on and on. This is the work intent cycle where the potential to change creates more potentials to change in the same manner to retain the appearance.

## The progress of cycles (shAnti)

Many of us who have grown up near temples, priests chanting slokas, been in one too many homas and been at many of the pujas for various different events that occur in our life and others lives know very well that many of these mantras, slokas, sthotrams end with "OM shAnti, shAnti, shAnti" and typically this is translated as

“Let us be at peace, peace, peace”. But, this is just a mangled up translation of a very beautiful concept of “shAnti”. This word like many other words in Sanskrit does not have a direct word-to-word translation in English or any other modern language.

To understand what this word really means, we want to look at the “shAnti mantram” to understand what shAnti can mean. There are many shAnti mantrams, but the below gives the best description:

*OM dyuH shAnti antarikShaM shAntiH*

*pRthvi shAntir ApaH shAntir roShadhyaH shAntiH*

*vanspatayaH shAnti vishvedevAH shAnti brahma shAntiH sarha shAntiH*

*shAnitir eva shAntiH sAmaA shAnti redhi*

*OM shAntiH shAntiH shAntiH*

*Translates to*

*The OM of the universe is shAnti, the space inbetween is shAnti, the earth is shAnti, anything obtained is shAnti, the turbulence is shAnti, the dense is shAnti, the energies of the universe is shAnti, the creation is shAnti, continuity is shAnti, shAnti itself is shAnti, time is shAnti, the OM in individual is shAnti, shAnti, shAnti.*

I have not translated shAnti here because as I have said, there is no word-to-word translation that conveys the meaning. But, when we take this in conjunction with the explanation in the section of “How we do the integral of derivatives very well” and look at things around us as they happen, we find that everything big or small starts off as a very minute unrecognizable piece of event.

For example the degradation of the road. It starts off as a small pebble on the road after it rains that wedges itself at the right hole in the road. As vehicles, heavy and light pass over this, the size of the hole enlarges slowly, piece upon continuous next piece, till it becomes a huge pothole. It starts off as enlarging slowly at first, but once it has reached a certain critical point we find it start growing really fast till it reaches some final pothole size from when on, it stays in that manner till manually filled out and fixed.

We can find this process followed in everything in reality. This process is termed as shAnti. The slow and sure progress from the start to the becoming to the become of anything that happens in the universe. Thus the above mantram tells me that the universe is also built in the same manner, step upon continuous sure step from the

smallest to what it is now, that which is in-between various astral objects is also built in the same manner, our earth is also built in the same manner, anything we get, any turbulence, any density, any energy, any creation, any time, the continuity we see, the individual, you name it is built in this manner. Slowly, sure step upon step accumulating to create this whole big thing.

The Surya Siddanta gives a very good picture of this process in chapter 2:

*vakrAyativakrA vikalA mandA mandatarA samA*

*tathA shIghratarA shIghrA grahANAmaShTaghA gatiH ||2.12||*

*tatrAtishIghrA shIghrAkhyA mandA mandatarAsamA*

*RjvIti paNcaghA jneyA yA vakrA sAtivakragA ||2.13||*

*tattadgativashAnnityaM yathA dRuk tulyatAM grahAH*

*prayAnti tat pravakShyAmi sphuTIkaraNamAdarAt ||2.14||*

*Translates to*

*Continuous curve connected to curve, consistently, slowly, very slowly, in this manner very swiftly, acquires eight in conjunction with path. There, very swiftly, in conjunction with obscure, consistently, slowly, very slowly, obtains a motion in the 5th, knowing this, that surely a curve accompanies a curve. Then, that regularly controlled path, if observed in accordance to the acquired, before being established that act of observation, gives the spark of manifestation.*

The question is on what is this concept of shAnti applied. This is also given in the same mantra we repeat over and over again. “OM”! So, if we take the indeterminate continuous Brahman which precedes the “activation / inactivation of data” and recursively apply the concept of shAnti to it then we should have the first cycle of active/inactive being started. Then recursively we can keep going to the lower and lower level cycles and keep applying the process of shAnti to the cycles, for example the work intent cycle. Then we should get the continuous step upon step progress of the work intent cycle to extend and expand the scope of the cycle. So, as more and more penetration happens, more and more data get activated, as more and more data get activated, the observer is able to observe more and more and hence slowly, surely step upon step pattern gets formed and keeps moving forward. This is how everything in the inner workings of reality progresses.

## The work cycle (Exploding the controlled part)

As we understand the work intent cycle, we start seeing a pattern emerge in the cycle that has been defined. For example, Mahadeva is the potential to change, Shakti is the detector or the change or actualization of a potential, Saraswati is an actualization of the change into a knowledge, while Brahma is a potential to create resource out of that logic, Lakshmi is an actualization of the resource and Vishnu is a potential to create equilibrium of those resources. So, a potential arises and an actualization happens of that potential and that actualization moves towards establishing the next step. Those actualizations have more sub-cycles in them to create a potential. This is incorporated as the work cycle.

The “work intent” cycle is the core cycle that works directly at the level of the Brahman, changing to activate “continuous bits into data” in the Brahman, include the activated data into the self that is formed, to keep introducing more intents to ensure the cycle moves forward and establishes an equilibrium. But, how does the “work intent” turn to work? How is this related to the scope increase or decrease of “work intent”?

Given that internals of reality has to do with establishing equilibrium, the logic formed is in a state of constant flux, adjusting itself to the changing state of the system. As the data changes, all the selves adjust themselves to the changing data so that all can keep going towards the same equilibrium. As I have explained before, because of the way the data is represented and the method in which learning is got, properties are associated and everything else works, all it requires is for the data to change, to change the logic and everything down the line, automatically and immediately. Again, as I have said before, each and every cycle is composed of lower cycles which contributes in the higher goal of equilibrium. The work intent cycle contributes into the top-most level active-inactive cycle. The work cycle is the next lower cycle that contributes into the work intent cycle.

As I have marked in the work intent cycle diagram above, we have two parts to the cycle. The controlled part and the uncontrolled part. It should be recognized that once a work intent is formed and the cycle started, the uncontrolled part of the work intent cycle will form come what may, it goes on to “become”. Control does not come back to the observer till the formation of Saraswati or logic. This is termed in Sanskrit as *karma* or the uncontrolled order of how

things are done. But, once the logic starts getting formed, then the observer gets back the control to observe or not observe, to form or not form the logic. This further controls how the next work intent gets formed. This falls in the controlled part of the cycle. Thus, the work cycle falls within the preview of the controlled part of the cycle and dictates how jnanam & para jnanam come together to create Saraswati or logic and continue further to produce more work intent.

This brings out another key difference between our coded AI and the inner workings of reality. If we look at the difference between the controlled and uncontrolled part of the cycle, we find that the primary difference is that one is "*a potential to be*", while the second is the "*actualization of that potential*". In the AI we code, we do not have a concept of potential at all. The closest we get to this is in reinforced learning where a logic is rewarded or penalized, but here the logic is formed and then decision taken to discard it or not. The reason we need to do this is because we are a logic-based learning system with a computational data representation and the AI coded are all offline data learning systems.

Given that the reality is a driven by a data-based system, it has the inherent advantage of data representation that allows for potentials to be formed inherently. This is the advantage of the OM data representation, the data need not be activated or inactivated. It has the potential to become an observed value. Given the strength of the "work intent", more continuous data bits can be activated or not based on the work done. And moreover given that it is a real-time data, it also has the added advantage of hardening patterns as it keeps recurring, allowing a more controlled creation of knowledge and intelligence.

So, what is the work cycle? The below shows the two cycles and their relation:

Introducing more intent to change and hence work intent
Forces of accumulation
generates work intent
jnanam / para jnanam come together
potential to change (Mahadeva)
The resources solidify and come together trying to establish an equilibrium (Vishnu)
Multiple paths interfere with each other to generate potential of more pattern
vijnanam or coherent knowledge is formed
As the path moves a pattern is recognized along the path (Work)
starts off an origin (which is one coherent knowledge set) that is persisted
Shakti to detect and observe change (Shakti)
Using the factory created by the factory of factories more intent in forward direction is formed (Lakshmi)
The path gets a direction associated with it
A moving apex gets formed from that origin to create more coherent knowledge set
A path gets established because of the detaching of the apex from the origin
logic forms the factory of factories to create resource (Brahma)
observed changes accumulate to form jnanam / para jnanam
many jnanam / para jnanam define progression and form logic (Saraswati)

The exploding of the controlled part of the cycle starts at the formation of the jnanam / para jnanam and links back to the cycle when more work intent is created to introduce more forces of accumulation. The basic concepts that drive this cycle are first: smara or recollection necessary to detect and recognize patterns and second: patterns can interfere with each other to form more interference patterns. As I have said in the explanation for Saraswati or logic in the previous section, logic is just coming together of the knowledge in a certain pattern. As I had given the example, the simplest pattern recognized can be similar to a sequence: “a linear increase in rate of change” followed by “a constant rate of change” followed by “a linear decrease in rate of change”. This can be considered a logic or progression of data. So, how is this logic created?

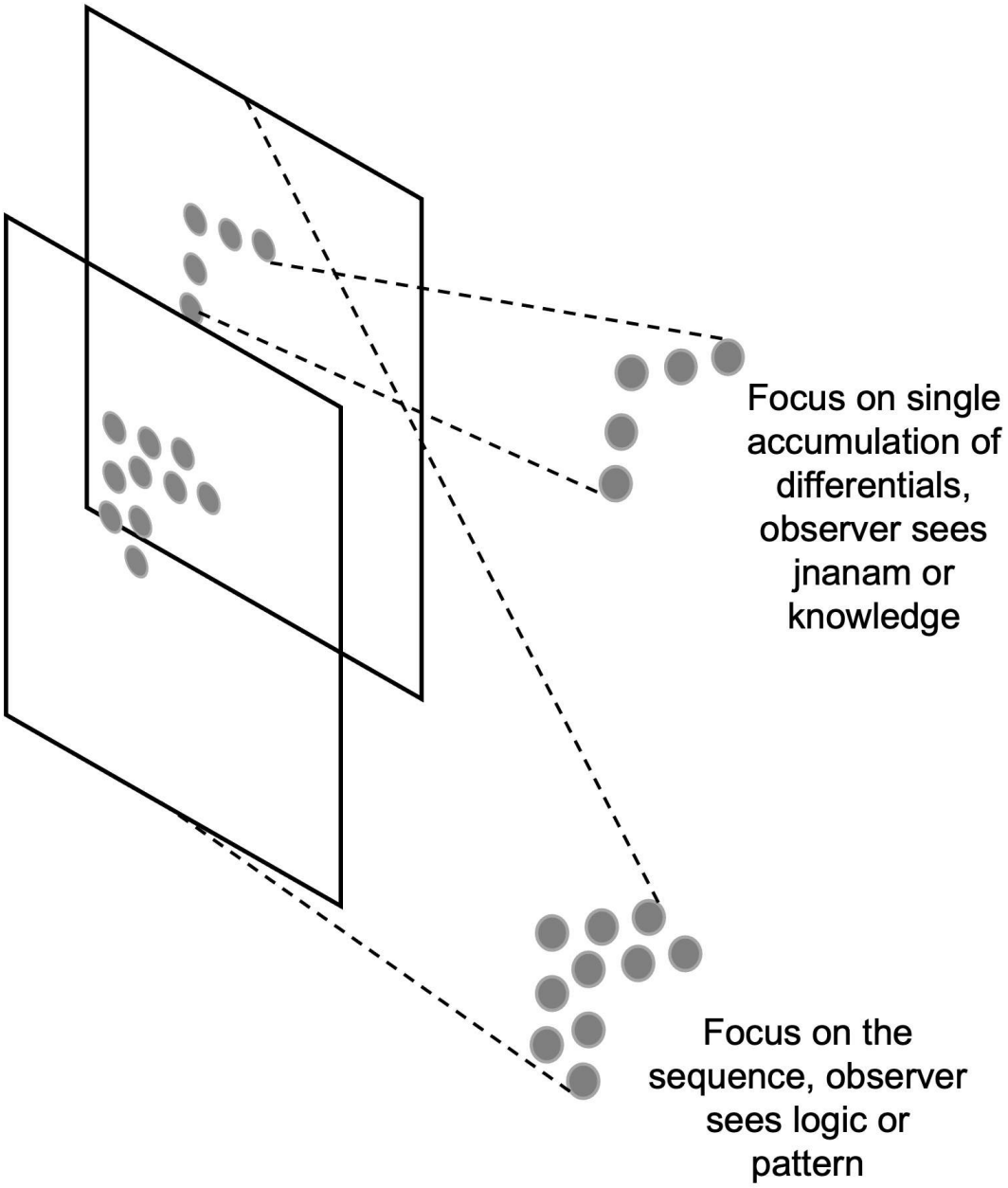
**Vijnanam or coherent set of knowledge**

The first concept that is necessary here is the concept of “vijnanam or coherence”. Once knowledge has been recognized, i.e., “a constant linear increase in change”, how can we start relating

multiple knowledge together to form logic? When multiple such knowledge come together, they form a coherent set of knowledge.

In neural networks, we typically leave logic at a single path representing a single knowledge learnt in the narrow domain band. We can consider the linking of multiple nodes achieving multiple activation functions on the same input data as it traverses from one level to the next as a set of coherent data. But, it is very constricted in neural networks to mathematical functions and the path as a whole represents just a single path of knowledge. In fact it maybe more apt to compare coherent set of knowledge to the “knowledge graph” that is present. There are graph neural networks that try to achieve the knowledge graph learning, but is very complicated when compared to what is present in the internals of the working of reality. By the simple fact of changing the data representation, we automatically get many required functions without much learning. So, how is this done here in the inner workings of reality?

As I have said in the explanation of Saraswati or logic, it needs to change the focus as shown below:

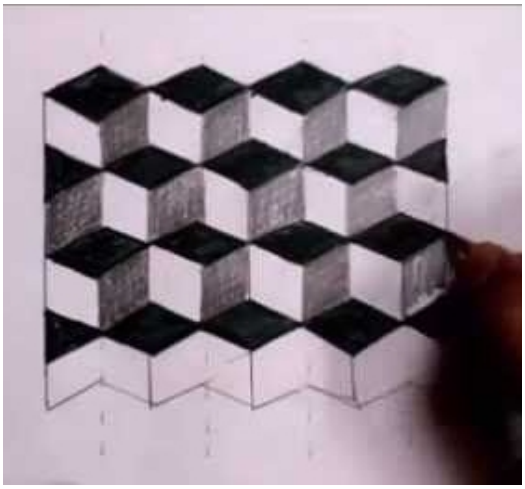

Focus on single accumulation of differentials, observer sees jnanam or knowledge
Focus on the sequence, observer sees logic or pattern

So, for logic to be formed the observer needs to extend its focus and look for repetitions. It should be realized that while lot of terms that is used here are seemingly very simple concepts, when applied in terms of an indeterminate or an unrealized or unknown set of data, as we are currently doing, they take on very complex meaning and need explanation and are no more so simple. For example, the concept of coherence is a very simple concept when we are looking at say two waves. We know they are coherent if, as we traverse the wave the phase does not change, or the amplitude, frequency and waveform of the wave remains the same throughout. Another example can be a set of coherent words. They are considered to be semantically coherent if a set of words can come together to convey a single meaning, otherwise they just remain a set of words with no

meaning or incoherent. These seem simple concepts because they are commonly understood to be this. Here, an already known understanding is applied rather an understanding being formed from scratch as we need to do when nothing is present.

But, when we look at a set of knowledge that is represented by data, such as “a linearly increasing rate of change” in the data set, what can be considered a coherent data set to this? If we applied a context to this, for example, we said we are looking for a saw-tooth structure, then it is possible for us to say, that the next data set I need is a “linearly decreasing rate of change”. Again, such a expectation is what is a “logic-based” system. Because a logic is driving the data. But, we are not looking at that. We have a continuous stream of data that is coming in and the algorithm is a data-based system. In such a system, coherence shifts from the model of description first to the data, to a model of data first to description later. This implies that if, the observer can detect the same pattern multiple times, then, the knowledge that go to making up that pattern becomes a coherent data set. How do we find the pattern? We come back to the foundation algorithm again here.

As I have said, the foundation algorithm is such that we can pick up the same algorithm and keep applying it over and over again to different constructs and we end up with the next higher construct. So, we saw that we could apply the foundation algorithm to the raw changes first to get knowledge or jnanam. Here, we find that we can apply the foundation algorithm to the knowledge or jnanam again, to get the vijnanam or the coherent knowledge. What is very very interesting about this is the fact that both jnanam and vijnanam are stored as just data. Thus, only that single action of folding continuous bits of data over each other, produces both jnanam and vijnanam, based on how the observer observes it. While we understand it as applying to the jnanam, it could very well have applied to the raw data. This again follows the compose-decompose philosophy. This can be very easily seen in an optical illusion. For example as shown below:



The above picture can be seen as multiple knowledge of black squares, white squares, light shaded squares, or we can put together "one of black square + one of white square + one of light shaded square" and create a vijnanam or coherent knowledge of a cube. The data for the three squares individually or the three squares together are the same representation, the data of three squares combined together or split. The observer has either tied them together or not. This is the start point of the work cycle. Multiple jnanam/para jnanam coming together in vijnanam or coherence.

**Data persistence and the promise of a pattern (smara and pramANA)**

In the computer world, persistence can be long term or short term. Long term persistence is achieved with databases or in files and short term with memory or memfiles. We need such a distinction in persistence of long term and short term because it is a logic-based system, where data is secondary. Given that processing is given prominence, without persistence a process can be executed to temporarily spit out an output which is lost once the execution stops. What is very important to realize here is, the computer exists in a world where "time" is a factor that needs to be contended with and kept track of external to the executing process. The processor tracks this using the clock cycle that is present within itself.

When we look at a data-based system where data is primary, data needs to be present for-ever for the system to work, because the data contains the knowledge that is used subsequently for everything else. But, an even more important point to note as a difference between the computer world and the internal working of reality is that while in the computer world, time is an external factor to contend with, in the internals of working of reality, time is a derivative of how the internal working of reality progresses. The

“working of reality” causes time within reality, rather than some regular external time driving the pace of reality.

As I have mentioned in all my books and blogs, “time” is not a concept that is even recognized by the Sanskrit literature. “kAla” which is usually translated as “time” in-fact should not be translated to time. It should be translated as “a series of changes” or that “continuous bits of related data strung together by some coherence and recognized by the self as a pattern”. The major difference between “kAla as a series of changes” and “time” is, that, while time is an external abstract concept to the changes and tends to be seen as a constant across all individual changes, the concept of “kAla” is internal to the pattern of the self and drives the progress of the self, hence each self can have varying periods of progression and no single global period of progression exists. The abstract for such a concept is got by recognizing “smallest recollectable change called vINADi” which varies for each and every self, but is an abstract that can be used to unify across all the selves.

When we look at it from these perspectives, first, we find that the data representation as a “continuous activated bit” in the whole truth is the driver for the formation of reality. *Hence this ensures that “the activated data” is available all the time within the scope of reality*. In the appearance of reality we exist in the latest rendered frame of the continuous frames of appearances we render for each of the series of changes in kAla. Thus, it seems as if the older data that created the previous frame is replaced by the next frame and hence we feel the need persist the previous frame states. We don’t seem to grasp easily the concept that all the prior activated data is actually present in the underlying truth, it is just the observer that is moving w.r.t the truth and hence it seems as if the frames are moving. To understand this, we can imagine an endoscopy that is done using endoscope. The endoscope is moved along the organ that is examined and what is visible is just the focus area of the endoscope, but the whole organ is actually present. It is just the visibility that is restricted because of the observer’s i.e., the endoscope’s form. Similarly, here, all the previously activated data bits are actually still present, it is just the visibility or appearance is only of the currently focused area of the observer.

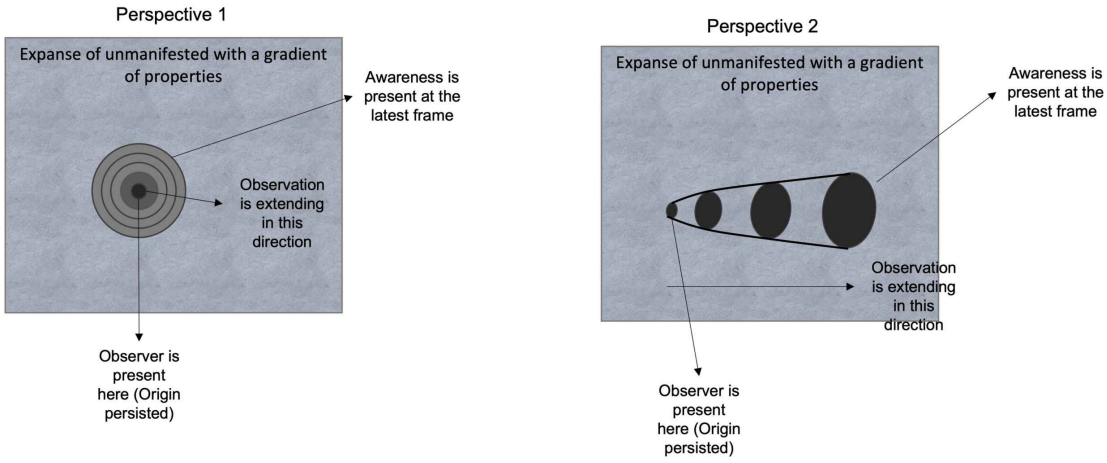
So, if all the data is always present in the truth why does the Sanskrit literature say that “smara” is the reason for the fifth dimension. So, what is persisted here? “smara” is typically translated as “recollection”. I have just given it the alternate meaning which is

the pre-requisite for recollection namely persistence. So, what needs to be persisted, if the original data is anyway always available?

It should be remembered that while the continuous bits of activated data are present, the observer has the capability to see the folded over versions which is the output of the foundation method applied across the board in the algorithm. Here, the start point or origin of the continuous activated bit from where data kept getting folded needs to be recollected so that the comparison and change can be computed. This origin then, is what is remembered and by doing so all the knowledge and coherent knowledge has started to form. This is different for different observers and hence, we start having multiple variations of individual knowledge at the same location and area observed by multiple observers.

The origin is a very important point to recollect, without which the reference is lost. For example, if we take a smooth uniform sphere and started rotating it, without a specific marked point on the sphere that indicates the point of start or origin, we cannot know when the sphere has finished one rotation. While, here it is not necessarily tracking the rotation, we are tracking progress and change which cannot be computed without the origin.

It is strange isn't it, that many concepts has many pre-conceived ideas the tends to blind us to what could actually be happening. The persistence of origin is also one such concept. If you read the previous paragraph, the idea conveyed is that the origin is a location that stays put, the sphere is rotating away from the origin. So, in our minds the idea conveyed is that the awareness is in the rotating sphere moving with the sphere while the origin stays and the awareness is observing the origin which is persisted. But, as we have seen through out, the working of reality is actually very very simple. Simple things seems to appear as if something else is happening. The simple fact is that we have unified the "*observer*" and "*awareness*". If we split these and say that the "observer" is always present at the origin, from which it observes variations of the activated data and projects the view as frames and the awareness is always present at the latest frame or apex, then we find that persistence is not really something that has to be accounted as to how it is done. It is the location where a capability exists to detect changes. Shown below:


Perspective 1
Expanse of unmanifested with a gradient of properties
Awareness is present at the latest frame
Observation is extending in this direction
Observer is present here (Origin persisted)
Perspective 2
Expanse of unmanifested with a gradient of properties
Awareness is present at the latest frame
Observation is extending in this direction
Observer is present here (Origin persisted)

I have shown two perspectives here, one from the front and the second from the side view as the observer extends its field of observation, though what is front and what is side view is very debatable, but I will leave it as this for easy understanding. Now, the observer is viewing the folded data from the origin and the amount of information present is till the last frame, hence the awareness is present at the last frame. Now, all the necessity for memory fades away, because we are always seeing everything from the origin, it is just that the awareness is always seen through the extended version. Hence we find that what we call memory is not always same, it is a recollection viewed through our experiences. At the origin is the Shakti or capability to detect changes, and as the truth moves its changes because of the forces of accumulation, the detected changes are viewed through those changes like a folded cloth accumulating and forming jnanam, vijnanam and persisting the origin.

Once, we have this concept of recollection of an origin and the data being available throughout the progress of an individual or self, now is when the Surya Siddanta says that pramANa is formed. It says:

*sasandhayaste manavaH kalpe jneyAshrcaturdasha*

*kRtapramANAH kalpAdau sandhiH paNcadashaH smRtaH*

*Translates to*

*Along with that transition into thought it appears to know four dimensions, by the appearance of making a promise, transitions to 5th dimension by persistence*

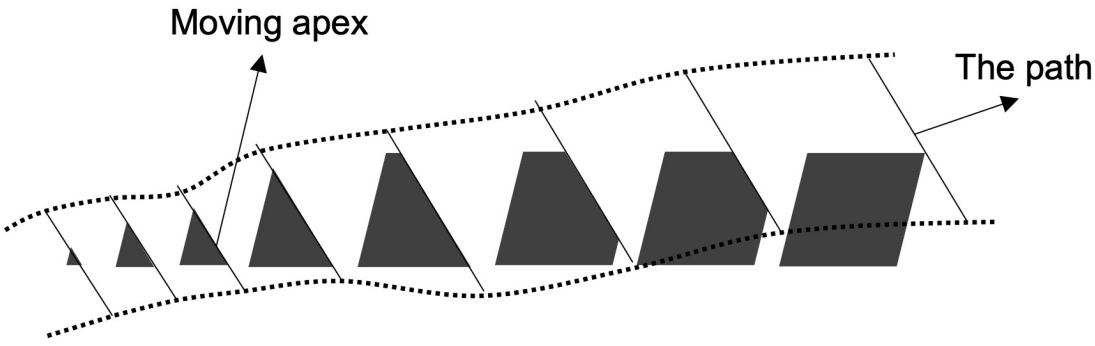
Typically pramANa is translated as “a quantity or measure”, but here I have translated as “promise”. In computer scripting languages such as node js or angular, a promise is a way to execute a function asynchronously which promises an output of the function in a

certain format. So, if an asynchronous function returns a promise<int>, then the function assures that by the time the function finishes execution it will return an integer. The promise or pramANA in the above verse is similar. A specific pattern of change combined in a specific manner is a thought. If this has a promise "*to become a certain pattern*", the fifth dimension is got when this thought is persisted. This thought then is the origin thought from which the promise to go further and become something starts.

**The moving apex, the path and the quality of direction**

Now that we have a persisted data and origin with a promise to become something, the question is how does it become that something. As we saw in the section on "the progress of the cycle", everything follows a process of shAnti or a unstressed, continuous step upon step progress towards something. Here also, we find that the starting thought with a promise progresses using shAnti. And by doing so, splits the origin into two. The origin which stays where it is and the apex that is moving. It is very important to understand this concept of moving apex.

In geometry we create multiple regular and irregular closed shapes such as circles, square, cube, sphere, parallelogram and so on. But, truly, if we look at the internals of the working of reality, and look at anything in reality, we find it is pretty impossible to actually have a closed shape or object. The reason we get a closed shape is because we have not put time into the mix. Everything starts out at some starting point and expands out based on the accumulation. For example shown below, how say a parallelogram builds out:




I have obviously not drawn it as a continuous growth, as it is in reality. I have drawn it as a set of discrete frames picked out as it grows out. Here, an apex moves revealing more and more of the form as more data gets accumulated around that starting minute

collection. We typically tend to envision an apex as a point or tip of some sort. But this need not be the case. An apex need not be any regular shape at all, it is just that latest something of a collection of the same thing. Here, I have drawn it as a line that is moving and shown some path it could have followed. But it as well could have been a 4D apex revealing multiple dimensions at the same time as it moves. Again, I have drawn this as a space image, which is also not the case in reality. If we took all of these and connected them together like a cut-out of a paper where it increasingly becomes bigger and can be opened out, that is what is happening, hence the parallelogram is just that set of connected data points becoming bigger and bigger as it accumulates.

If the verse I have listed in the previous section is to be understood, then what is accumulated and revealed is a set of thoughts to form a group as the apex moves. Like we were talking about jnanam and vijnanam in the prior section, those thoughts revealed form a set of coherent knowledge. More and more thoughts are clustered together onto the first thought with the latest set of coherent thoughts or vijnanam forming the apex. If, as this apex moves we start looking at the connected version of the apex, we find that it has followed a path. This is the path of existence of that set of thoughts as it grew from that one single thought created by connecting continuous bits of activated data to a series of coherent knowledge. Also to be understood is that the path cannot be as straight and in an open path as I have shown. After all what is straight, when we are looking at an expanse with no reference?

When we look at a expanse of the Brahman with no reference point other than an origin that is persisted and we look at no concept of space or time at all and the series of changes establishing themselves are the only existing referential ideas, then there exists no concept of direction, straight, open or closed. Because all of these need more points of reference. But, when, because of the moving apex and a path, more and more accumulations occur, then, the direction in which there is an increase of accumulated thought becomes a forward direction and the quality of direction gets associated to the self. Till then, it has no notion of direction at all. Which gets established first, the path or the quality of direction, is something we cannot decipher, because as one grows the other also gets established. Though, the BHPS says that the association of the quality of direction is the reason for starting to limit the features that can get associated with this self. It says this:

*vyaktAtmakastribhiryuktaH kathyateanantashaktimAn*

*sattvapradhAnA shrIshaktirbhUshaktishca rajoguNA*

*Translates to*

*By connecting the nature of manifested to the directional becomes, restrains the rendering of the infinite capabilities, the true essence of the natural state is mixing the becomes capabilities and the penetration capabilities and is a quality of dopant*

Thus, we start off with a dopant that has the infinite potential to go any direction and become anything in that expanse of the Brahman or truth. As the accumulation of bits of activated data occur, we have started reducing that scope of that dopant from that infinite to what is adopted by the accumulation to become. Then due to the penetration of that dopant, a direction starts getting associated and subsequently the rendering gets restricted to that which has been established. So, how does the forward direction get established? This is explained in the Surya Siddanta as:

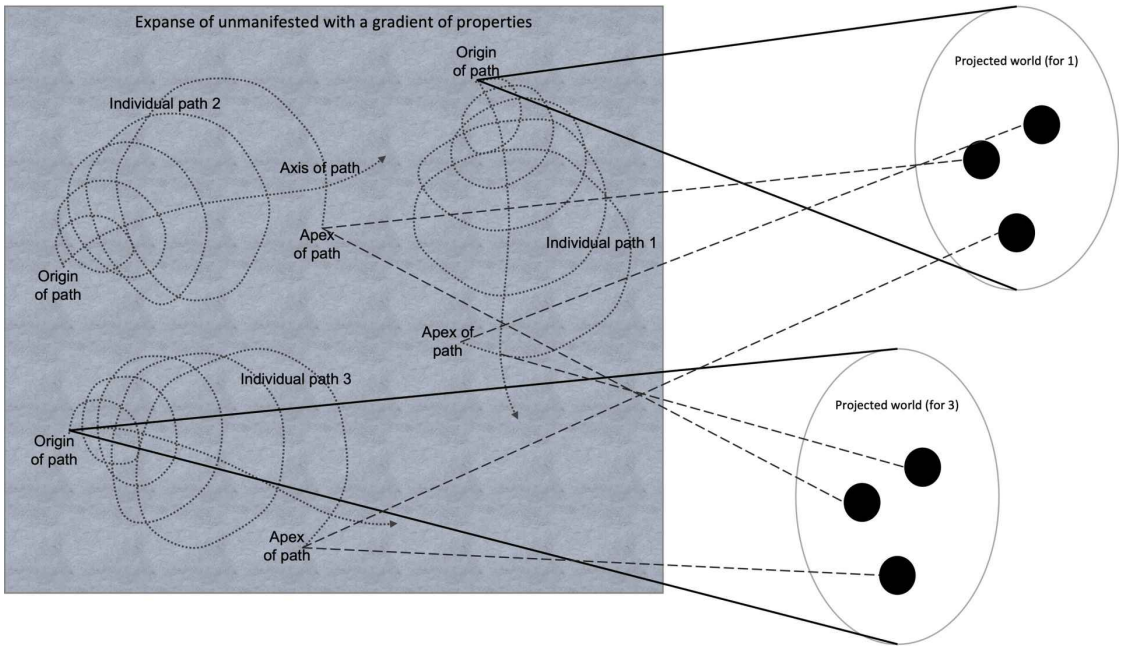
*prAggatitvamatasteShAM bhagaNaiH pratyahaM gatiH*

*pariNAhavashAd bhinnA tadvashAd bhAni bhuNjate*

*Translates to*

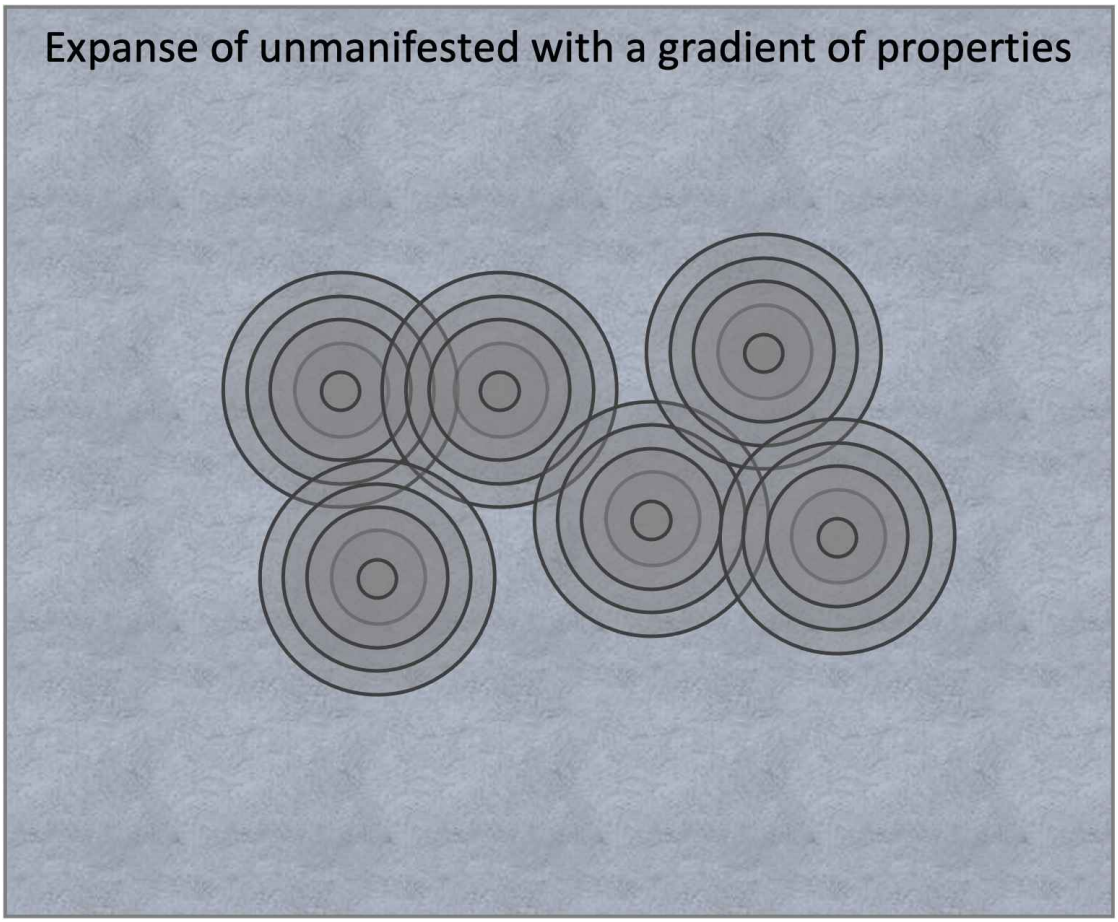
*Seeking other thoughts within the series of becomes in the forward path, is path of each "I", origin detached from expanse, from that origin emerges experience*

So, that single thought starts seeking out other thoughts within the series that matches itself for coherence. As we see from the penetration concept that I explained earlier using silicon, a free electron can only move if there is a hole or a deficiency in electron in one of the bonds. Thus, even though to it seems as if electron is flowing, it can only flow because of the existence of the molecular bonds with holes. This can be seen as just that, electron occupying another hole, or can be made into a generic logic, where the change has sought out another location in the expanse that matches the logic that it has learnt. Such a movement of change is considered as seeking. This detaches the apex from the origin and a path gets established and an experience emerges. This can then be projected into worlds that are created for each individual. A representative diagram is shown below (I have explained more on the projected world in the next chapter):




## The pattern, interference and work

As I have explained above, “work intent” is the potential to be able to activate enough continuous bits of data to form that energy pattern. So, work is a realization of that “work intent” i.e., the realization of the pattern that had a potential to form. When we look at the start of the intent at the origin, it starts at a standard pattern that makes up thermal energy. The “work intent” formed is for the potential to form the pattern associated with thermal energy. But, as the intent progresses forward, it encounters the potential to form other patterns when it hits patterns in the forward path of other intents. These multiple patterns cause an interference. The observer at the origin can distinguish the difference in patterns and interference with other patterns. The interference pattern now forms the bond between the various intents as shown below:




Once the bond is recognized and formed, the "work intent for that interference pattern" is created i.e., a potential to form more interference pattern, thus forming more "work intent" which then continues the work intent cycle. To be noted, I have called it interference. In Sanskrit it is called "dyu".

**Extension and expansion**

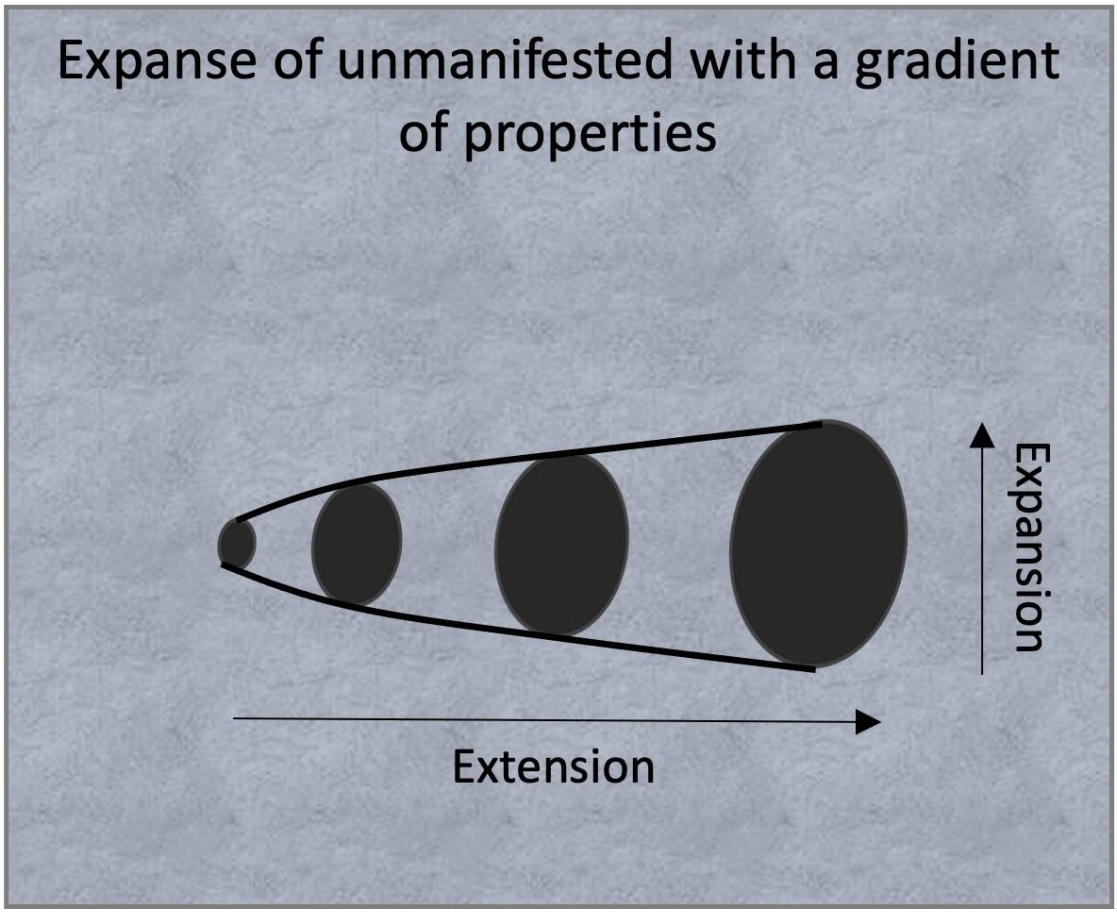
Once a self is able to observe itself at some origin or reference point, it moves from that origin point by penetration as I have explained before. It can grow two ways, extension and expansion. But, the Brahman is a continuous indeterminate in all directions, while I say directions again as I have said before, that is because there needs to be a way to explain. But, it should be understood that direction is also only because of an observer and an establishing of a forward path because of accumulation. So, what this statement really says is, if there is an observer, then every direction the observer travels it only encounters a continuous indeterminate Brahman. Then, how can a distinction be made between extension and expansion?

Again to be understood is that the description of the observer is what distinguishes the growth as two ways. If no description were

present, this would not really be the case and it would just be. When we talk about the work intent cycle we find that we have two ways in which the cycle can increase in its progression. It can either activate new data from the indeterminate whole and thus change itself to progress with the change. This is extension.

The next way in which it can progress is by expanding into other jnanam and knowing more and more para jnanam i.e., by increasing the field of vision of the knowledge. This is expansion.

With extension, the self is increasing its dimension of "time" as we call it, because the "series of changes" included within the self has increased. With expansion, the self is increasing its dimension of "space" because the "view of the remote has increased".



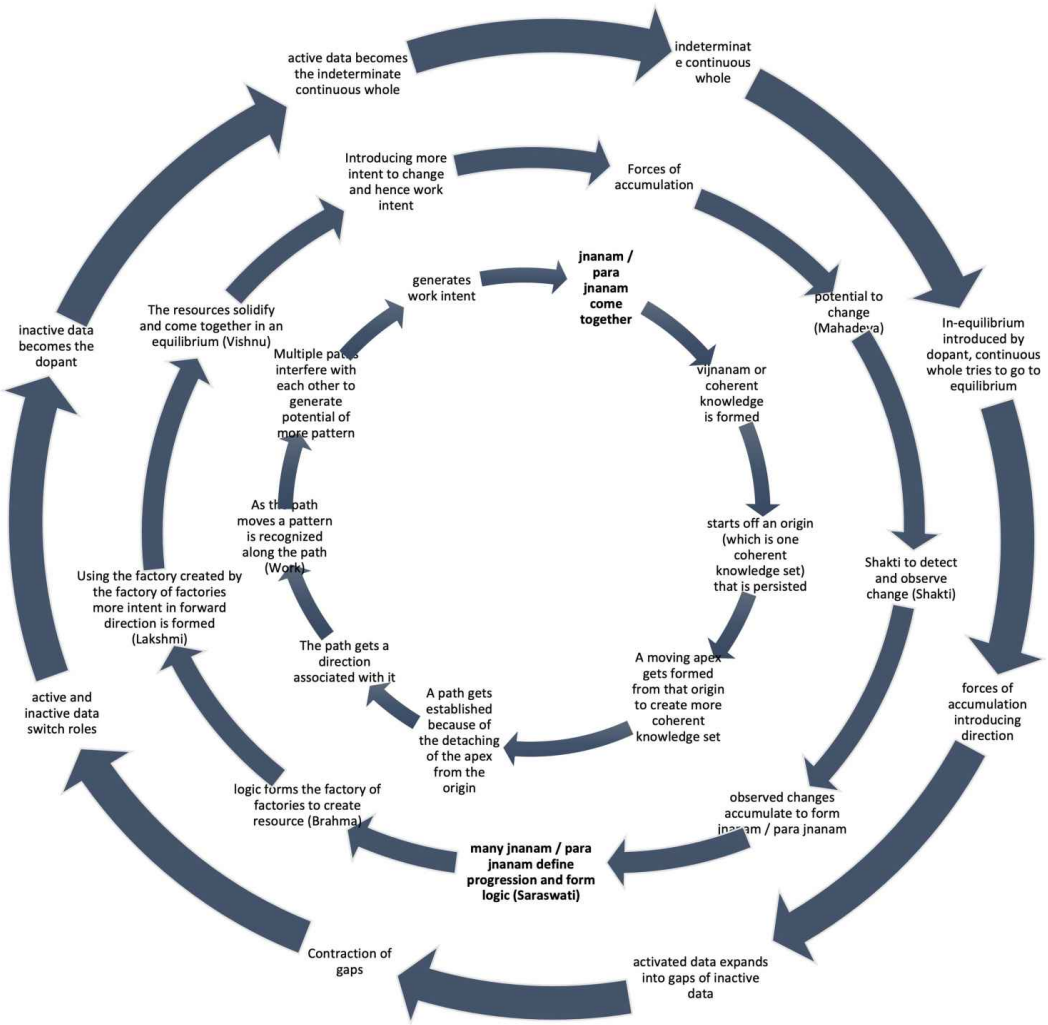

But, irrespective of which type of increase is handled, extension or expansion, both follow the concept of shAnti to increase its scope, continuous step upon step and hence progress the self.

## Tying it all together into an algorithm

Isn’t it so strange, that all that is done is so simple: change the data representation, associate it with inherent states, add continuity to the data, introduce a single agitation to change the data expanse so that it tries to come back to equilibrium, observe changes introduced in the data across the continuous data from different angles with many observers by just folding over one continuous bit of activated data over the other to achieve the observation of accumulative differentials and recursively observe these accumulative differentials of the intermediate data sets to infinity and keep extending and expanding the observation field, we get this whole big reality. To this add various appearances and we get this observable universe around us!

The simplest embodiment of reality is seen when we take a fluid stir it up and stand it and allow it to come to rest. If we can associate awareness to each and every macro and micro level elements that make up the fluid, so that they can create imaginative appearances, reality is simulated. The strangest part of it is “what is awareness”? Just by being able to “detect the change introduced because of stirring and take on the overall goal of coming back to rest, the elements have automatically become aware of the whole”!! It has never been said that awareness should be what we as humans call awareness. That is just a description and understanding given by humans! Awareness is knowing your surroundings and being able to react to changes in it, which is what each and every macros and micro elements in the fluid is doing. So, just stirring and letting the fluid stand creates this whole universe of appearances. We could give it many descriptions of cycles of existence, energy, prime energy, capability to detect change, logic, work intent, work, awareness, consciousness and so on. But, all of these are just descriptions of what is inherently happening.

This is the most simplest and *THE* most beautiful algorithm that I have seen in my short life. Given below the three cycles that are occurring.


active data becomes the indeterminate continuous whole
indeterminat e continuous whole
Introducing more intent to change and hence work intent
Forces of accumulation
jnanam / para jnanam come together
generates work intent
potential to change (Mahadeva)
In-equilibrium introduced by dopant, continuous whole tries to go to equilibrium
inactive data becomes the dopant
The resources solidify and come together in an equilibrium (Vishnu)
Multiple paths interfere with each other to generate potential of more pattern
vijnanam or coherent knowledge is formed
starts off an origin (which is one coherent knowledge set) that is persisted
Shakti to detect and observe change (Shakti)
As the path moves a pattern is recognized along the path (Work)
Using the factory created by the factory of factories more intent in forward direction is formed (Lakshmi)
The path gets a direction associated with it
A moving apex gets formed from that origin to create more coherent knowledge set
forces of accumulation introducing direction
active and inactive data switch roles
A path gets established because of the detaching of the apex from the origin
logic forms the factory of factories to create resource (Brahma)
observed changes accumulate to form jnanam / para jnanam
many jnanam / para jnanam define progression and form logic (Saraswati)
Contraction of gaps
activated data expands into gaps of inactive data

Can such an algorithm coded into an AI? I sure hope we can do it. But, I find that using our data representation and the limitations in the processing that is possible, the simple algorithm becomes hugely complex to code. Even the simplest foundation method which gets automatically realized because of the data representation here, becomes very complex to implement with discrete data sets. I believe a better approach would be to use hardware to achieve a few of the pieces of the algorithm rather than code it all in software, because continuity of data is something that cannot be achieved in software at all and has to be a hardware up implementation that uses the appearance due to the continuity rather than using the continuity directly.

## A highly efficient real-time search algorithm

So how does this come useful when looked at from the usecases addressed in the AI algorithm? What we require as an usecase in AI is to be able to apply a logic to a given input data set and retrieve

relevant knowledge and intelligence associated for a given dataset. As I have said one of the major difference between the AI algorithm we code and the working of reality is that the goal of the AI algorithm is to learn and store the logic in a learning dataset while in reality the goal is really to (re)establish an equilibrium in a dataset that has gone into a state of inequilibrium due to an introduced agitation. Thus, while the AI algorithm that we code stops at a certain state subsequently listening to input data to apply the learnt logic, the working of reality continues perpetually switching between active-inactive cycle. So, how can this be used to give meaningful output that we expect of an AI engine.

There are many applications of AI in the current landscape. Not all of them require a full-fledged AI or in-fact many do not require AI at all. For example, the "autonomous car" use case or the "robot for vacuuming the house" and many such use cases does not require a full-fledged AI. What they require is an algorithm that allows them to learn a skill. But, use cases such as using the medical history of a number of patients and being able to identify patients that can potentially be at risk of having a medical condition or detecting that they have a medical condition, or a use case where discrimination in insurance claims or job applications can be eliminated requires knowledge and intelligence based systems.

So, what is the difference between the first set of use cases and the second set of use cases. The first set of use cases require skill while the second set of use cases require knowledge and intelligence. Skill is an ability to do a task very well, while intelligence is one where a conclusions needs to be formed based on prior knowledge and prior relations found in knowledge put together into an intelligence. While skill and intelligence develop over time as more and more experience is added to it, they do not require the same type of learning algorithms. Skill is knowing the most apt reaction to real time data that is continuously occurring to the object while the intelligence is a conclusion that needs to be reached based on comparison of the data of a given situation to other known data in a similar situation. For the second set of use cases use cases where matching knowledge and intelligence needs to be found to make a conclusion, a search algorithm on the churning environment will give the knowledge and intelligence.

The difference is in the way we view the usecase. When we have a narrow domain specific logic, then, that logic needs to be run on the input data to get an output. This is how a logic-based system works. But, when we look at a data-based system, the use case needs

to be recast to a different version, because everything we have is anyway data. All we need to do is search for that most relevant data that matches our dataset in the whole dataset we have and we will get the knowledge and intelligence associated with it. The advantage of such a redefinition of the usecase is that we will achieve the commonly missing functionality i.e., where we can adapt to new situations easily and apply a known solution from the most relevant situation and build and classify the current situation into a different class as we gain more experience in it. So, all we need is an really efficient search algorithm in the vast expanse that is going towards an equilibrium.

As I have said previously, a data-driven algorithm is one in which the data representation is such that it actualizes an algorithm. I already have explained how a highly efficient real-time foundation algorithm of integral of differentials is realized just because of the way the data is stored. The next major real-time algorithm that is realized because of the data representation is a “real-time search algorithm”.

We find that the type of searches that we can do on the data that we have collected over years of experience is in real-time and the types of adaptation and the various analysis we can very easily do on this data is very difficult to achieve in our computer based search and analytics algorithms. We can very easily pull up relevant data upfront, even though we have a huge amount of data that we need to process to pull it up. While in computers we struggle to pull up relevant data even with various complex search criteria. Even if we did pull up relevant data, we find that the speed of execution of the search varies based on the amount of data present in the database and the number of tables we need to pull the data from and so on, that on large data sets having an application that does a real-time search is impossible. It should also be noted that in computers that the kind of results that we search in a database in real-time is very limited. To do any kind of meaningful data searches, we should have pre-processed and stored the data. In these cases we can only pull what has been pre-processed. While we can very easily search a vast data set of a whole lifetime of experiences to pull up all or most disparate relevant and related data. The type of search that we are able to do is explained easily by the “cocktail party effect”. Wikipedia explains this as:

*The cocktail party effect is the phenomenon of the brain’s ability to focus one’s auditory attention on a particular stimulus while filtering out a range of other stimuli, such as when a partygoer can*

*focus on a single conversation in a noisy room. Listeners have the ability to both segregate different stimuli into different streams, and subsequently decide which streams are most pertinent to them. Thus, it has been proposed that one's sensory memory subconsciously parses all stimuli and identifies discrete pieces of information by classifying them by salience. This effect is what allows most people to "tune into" a single voice and "tune out" all others. This phenomenon is often described in terms of "selective attention" or "selective hearing". It may also describe a similar phenomenon that occurs when one may immediately detect words of importance originating from unattended stimuli, for instance hearing one's name among a wide range of auditory input.*

While the article only emphasizes on the the fact that we have "sensory memory" when it talks about parsing stimuli and identifying salient information, we find that this is not true only about sensory perception, but also true of our own "internal memory". We can very easily filter out the salient or relevant experience from a lifetime of experiences and adapt it to fit into the current situation. Isn't it strange that while we have used this extensively and seen it being used extensively not just by us and others "who we consider not intelligent", but also the animals, plants, and many inanimate things in various forms, we still are not curious to know how this can be achieved so easily, while we with all our complex representation of data cannot do it in our invented instruments?

As I have said, the algorithm is visible even in "so-called inanimate things". Let's take the flow of a river. We all have seen rivers in India being polluted absolutely with no regard for the river itself or the fact that the waters of the river was the reason that human life along the banks of the river have survived for so long. The very famous Ganga river's pollution is well known and many attempts have been made to clean the river, though unsuccessfully. But, why do I talk about a river flowing instead of a search algorithm, you ask? It is simply this, if we follow the river's course, as the river flows, we find that it becomes cleaner and cleaner and after a few kilometers of flowing, the river has cleansed itself of most of the pollution. How does this occur? This is a perfect example of how data search and filtering works in reality using "the forces of accumulation". I have explained forces of accumulation before w.r.t the work intent cycle where the work intent cause "forces of accumulation" creating change. Here, it is applied at a higher level to be able to search and filter. We find that "forces of

accumulation" is a generic concept that can be applied on many properties. Here, the "forces of accumulation" separates matter by its fluidity. The fluidity of water being different from other pollutants in the water, water of the river becomes cleaner as it flows away from the source of pollution.

This is a visible, natural example of the "forces of accumulation" in action. The other examples of the "forces of accumulation" used for search and separate can be seen in action, in the act of winnowing that separates a mix of multi-weight, multi-size matter by their weight and size, the action of sieving that filters a specific size matter from a mix of multi-size matter and so on, these being human aided invocation of "forces of accumulation".

Given that we know that the "forces of accumulation" acts when a data is introduced into the system, we just introduce the data into the field of vision of the observer and observe via both the current state of awareness and the data input. What is observed should be the most matching differential into which the input data has accumulated. So, as I have said time and again, just because of the change in data representation an algorithm is automatically realized which does not need too much of processing and is inline with the working.

# How does this translate into this reality around us?

You must be saying all this is fine. There can be data, there can be Mahadeva and Shakti and progress as shAnti! But how does that all become this world that I can see, touch, feel, hear and is very difficult to break-out of? How does it become this world with seemingly infinite space and infinite time which I have thought of as a space-time continuum. A physical, material world that seemingly follows a certain set of rules that I understand in terms of science such as physics, chemistry, biology and express as mathematics. A physical, material world where I have a body, fall sick, have emotions and reactions. Where I have to earn to live. Where I have to interact with other people and keep them happy, keep myself happy, not fall into any traps. Oh so many different things to take care of right. How does the description of the inner workings translate to this world around me? The Bhagavad Gita has explained this very correctly:

*vyavasAyAtmikA buddhir ekeha kurunandana*

*bahushAkhA hy anantAsh ca buddhayovyavasAyinAm ||2.41||*

*yAm imAM pushpitAM vAcaM pravadanty avipashcitaH*

*vedavAdaratAH pArtha nAnyad astIti vAdinaH ||2.42||*

*kAmAtmAnaH svargaparA janmakarmaphalapradAm*

*kriyAvisheshabahulAM bhogaishvaryagatiM prati ||2.43||*

*bhogaishvaryaprasaktAnAM tayApahRtacetasAm*

*vyavasAyAtmikA buddhiH samAdhau na vidhIyate ||2.44||*

*Translates to*

*The intelligence of the intent self is just one Arjuna, the efforts of that intelligence is many branched and endless. The moving flowery description of this is rendered when looked upon by awareness, by this knowledge and descriptions Arjuna, there is nothing else other than this description. The heaven of the thoughts of want is remote, given as a result of the produced work, the character of many actions is the path of experience. By the continuous experience, the movement got fascinated by awareness, does not establish the uncertain homogenity of the intelligence of the intent self.*

From the previous chapter, we know the intent self is the one that drives all the various cycles that are the foundation block. So, when we look at that intent self rather than this many rendered around us, we can start seeing the homogeneity in the way the appearance reality works around us. Isn't it strange that this world that is feels and senses so complete and in of itself is basically just a description of that inner workings of reality that is rendered?

But, what is more beautiful to me is that the completeness is such that every experiment, every thought and every action feels so isolated and working independently by itself just in this descriptive world without even a flicker or inkling of that inner world working wonders inside. We are all happy and content to think this world around me is what it is and search for the answers to our questions also only within this world and are satisfied with the results we find of the experiments without even understanding that those experiments are designed to only cycle back and give us the results in this descriptive world. The beauty of isolation should be like this! And, here we are talking about an app on the phone that connects to a server backend and all it requires is a small hack to expose the existence of such an architecture and lay bare all the code and logic that has gone into its working.

The beauty of this is such that even after we are given the description of how this whole thing works, it is very difficult for us to see, accept and understand what is said in those texts. The beauty of the simplicity is such that when we do understand, we look at it and say "Really, is it just this? Naw, it can't be! There has to be more to it!!" And yet, we cannot recreate the complexity that has evolved from that very simple algorithm! That is the beauty of this isolation of description from the actual working! I have searched high and low to find how this continuity in data representation can be achieved in our descriptive world and I find, it cannot be achieved in isolation as I have described in the data representation section, all I can do is allow the inner workings to take care of it and I just access the result appropriately.

When we read a number of Sanskrit literature, we find that the explanation for any of this kind of reality, i.e., a reality of materialistic appearance is not addressed in depth at all. The Bhagavad Gita talks about the "deha" in just a manner of tolerating the existence of the form and not trying to find a solution for it. For example:

*dehI nityam avadhyoyaM dehe sarvasya bhArata*

*tasmAt sarvANi bhUtAni na tvaM shocitumarhasi ||2.30||*

*Translates to*

*That which contains the body is constant, that one that exists in the body of all is indestructible Bharata. Hence don't regret thinking about the limits of all and the limits of the becomes.*

The Upanishads rarely mention "the material body" and more talk about forms and rendering. When we read it, we need to ask ourselves why this is so? There can be many reasons for it. In the beginning my thought was that some form of control of the rendering can be got by controlling the underlying layers and this is what the BHPS was trying to explain in astrology. As I have proposed, I believe jyothiSha is not really astrology the way we understand it, but should be translated as "jyA + eSha = control + seeking". So, my theory was that if the distribution of the seeking was controllable, then the world and reality around me is controllable to the extent of being able to specifically control the events occurring. My take was that this control would have transferred all the way to the rendering. But anymore, I wonder if that is really the case. Control cannot be got from the rendering. If it were, atleast a mention needed to have been done to that effect. The other reason could be because of the way every thing evolves. It can be that, at the time the Vedas and the Upanishads were written the fifth dimension was just evolving which means the body or form as we observe it should not have existed. It should have been in the evolving state. Confused? Let me explain.

## Appearances and actuals

We need to ask ourselves what is this reality around us as we understand it? The answer is actually, that which is considered as science fiction and laughed upon by everyone. It is like how it is said in Shelock Holmes. When you have eliminated all which is impossible, then whatever remains, however improbable, must be the truth. Reality as we understand is purely a simulation or an illusion that is created by our own brains. It is important to note that I have said "*reality as we understand it*" not reality is an illusion. What does this mean?

If we consider the observations that we collect around us using our sense organs and try to explain it by creating more thoughts that are formed because of the input from these sense organs, that is "reality as we understand it". For example, the touch that we "feel"

when we come in contact with an object such as a table or a chair is just a type of thought created within us because of the signals generated by the touch. The essence that is created comprising of images, sound, smell, taste, touch etc., that is created because of a series of inputs from the appropriate sense organs is also just multiple thoughts consolidated into a single essence thought. These in the Sanskrit literature is called as “bhoga” or “experience” related thoughts.

Isn’t it strange that we have not stopped to think that, “while the image that I see, is just a stream of light reflections on the object, from which I have created the image, I cannot see that light stream as is, as it was input into my brain and I can only see the image formed?” Or even stranger still “when it is just a stream of electrical signals that are sent from my eyes to my brain, how is it that I cannot first see the electrical signals as is, but just see the formed image?”

When we ask the question in this manner, we start asking ourselves, “what is it actually, that on which the light has reflected?” *Do I know at all?* In-fact the question should be “*Can I know at all?*” If everytime light reflects on it and falls on my eyes, my brain translates it and gives me only the image, how can I know what the object is on which light has reflected? We can surely add a external device such as a camera to capture the reflected light, but then again, it also can only show me the reflected light, not what that on which the light reflected!

We believe the answer to be matter composed of atoms or even still smaller particles of photon. But, the question here is, if light stream is falling in my eyes which is translated to electrical signals, my brain is receiving these electrical signals, but is giving me back an image, then what prevents the same from happening for an “atom or a quantum particle?” There are more levels of transformation here. There are signals received from any of our instruments, which is presented in some manner to our sense organs, which then gets converted and the brain creates a thought for what it is. How can I ever be sure of what is translated from the instruments to what is given to my brain to that what is given back by my brain, any of the actual that for which the brain is giving back the thoughts is retained?

I gave the above as questions so that we can enter into a state where we start asking the real question “what really is reality?”. It is only when we are able to break the brain away from the resounding

belief that what we sense is reality and this question occurs to us naturally, will we be receptive to understanding the difference. The Dakshinamurthy sthothram says this very aptly:

*Viswam darpana drusya mananagar thulyam nijantargatham,*

*Pasyannathmani mayaya bahirivoth bhutham yatha nidraya,*

*Ya sakshath kuruthe prabodha samaye swathmanamevadwayam,*

*Thasmai sri guru murthaye nama idham sree Dakshinamurthaye.*
*|| 1 ||*

*Translates to*

*The world is a reflection of the truth on the mind's mirror as if the atma is watching its own projection on the outer world just as a dream during sleep. I salute that Dakshinamurthy who by virtue of action lets the atma become aware of time and experience duality with only one's own self.*

To look at this from the computer parlance, this reality around us can be considered to be equivalent to "the UI rendering on browsers and other apps to display the data" while the actual logic to process the data is present at the server". Similarly the actual meat of this whole reality with its data and logic and everything is a process that is progressing in the environment created in the truth that sustains this rendering. This environment and its working is what I explained in the previous chapter. What we understand as reality is in fact just the rendering of that inner working.

It should be again noted, that while the "rendering of a UI in computer" is very simplistic, this is not the case in reality. One important point to note is that all Sanskrit literature has a specific theme for appearances. "vag or vaca or varNa or ucyate" and so on are some of the terms used. While in the standard translation vag or vaca is translated as speech, I believe it has to be a more generic term, namely, "description". Rendering is caused because of description of that underlying something by that underlying something. Hence the utter conviction that what is described is all there is.

Say, we take jnanam, which is the accumulation of changes observed by Shakti and describe it, for example as "a pattern of change in the form of a guassian curve", this is now a description of "jnanam" and becomes one rendering for that "jnanam". So, we have "jnanam" as the actual "collection of data", while the appearance of it is "a pattern of change". Another example is for

“intent”. The typical form that is described for it is “minA rAshi” or “multitude shaped as a comet shape”. This then becomes a description for what is represented by the underlying truth as “intent” which is again just a collection of data. Thus, we have “multitude of jnanam coming together as integral of differentials” as the actual intent, while the appearance of it is “the comet shape”.

According to the Sanskrit literature, we could take anything which I described in the previous chapter and if that anything is able to describe itself then it becomes a rendering. Rendering is not limited to what we see as the “material body” and this is not what is described in Sanskrit literature. To be noted also here, when we typically understand “describe” as using a language to express something which is not the same “description” talked about here. “Description” can be in any form. For example, “a rut formed” on the road because many vehicles travelled on the same contour over and over again is also just a “description” as I have explained before. Thus, we have an appearance and an actual which is merged into one understanding, that is the appearance for us. We need to split them into two parts and understand them as two separate, one concept the other a rendering of the concept only then we can start seeing the algorithm of the universe for what it is.

**What is cognition or awareness?**

As I have been describing in the previous section, there is always two parts to understand when we look at reality. The appearance or description and the actual. Cognition and awareness also has two parts to it. So, what is consciousness or awareness?

The concept of two-part is bought out in BHPS as “twin-produced” or “dvija = dvi + ja” as a part of describing “budha”. While, “budha” is standardly translated as intelligence, it also has a meaning called “awakening”. Here, it is used as awakening, which then is a description of awareness. It says the following about this:

*raktashyAmo divAdhIsho gauragAtro nishAkaraH*

*nAtyuccANgaH kujo rakto dUrvAshyAmo budhastathA ||3.16||*

*gauragAtro gurujneyaH shukraH shyAvastathaiva ca*

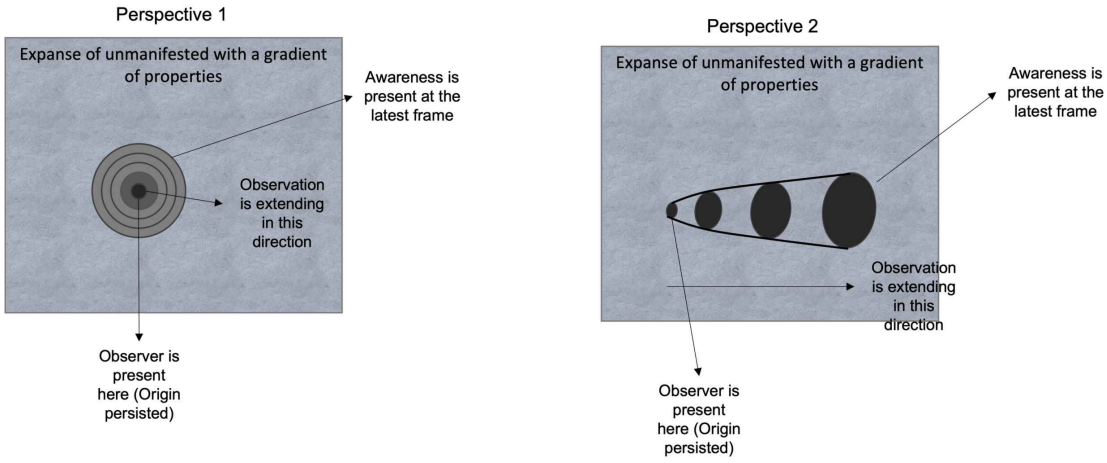
*kRShNadeho raveH putro jnAyate dvijasattama ||3.17||*

*Translates to*

*Clustered assembly attached to divisions of intention with a pure moving tail without multitude, awakening exists attached to the moving changes of not the tip most part of curve of the clustered assembly. The density of the pure moving tail exists by knowing the potency of the moving cluster of these, and the knowledge of the appearance of at-rest subsequent to turbulence, supports the first of the twin-produced*

The description here is a continuation of the intent which according to BHPS is a miNa rAShi or a multitude with a comet shape, which has tail. When this is viewed without multitude, but as one, then there exists awareness not at the head, but at the tail of the comet shape. What is important here to note is a single very important point. As I have been saying all through the document, words invoke many pre-conceived meanings for us and forms a barrier to the correct understanding.

Here, the word because of which the meaning is lost is “shape”. Typically shape is associated for us with an actual presence of space and a form. While, I translated it here as “comet shape”, this is because I wanted to convey the meaning with minimum words. What is important is to know the if we take many related “thoughts that are root thoughts which purely are indicative of change” and cluster them together, they become an assembly. Rather than thinking about same strength thoughts coming together into the assembly, say we had many different strength thoughts clustering together, the head or start of when the intention started to form keeps getting reinforced as more change thoughts get added to it. Thus, similar to the light intensity in a comet, we have higher strength thoughts at the head and lower strength dimming thoughts at the tail forming the comet shape. So, rather than a space related shape, we are looking at a thought strength related shape. Now, when we read this we find that according to the verse, the awareness exists at the tail end rather than the head end. This is similar to what I had described earlier w.r.t origin, shown below:


Perspective 1
Expanse of unmanifested with a gradient of properties
Awareness is present at the latest frame
Observation is extending in this direction
Observer is present here (Origin persisted)
Perspective 2
Expanse of unmanifested with a gradient of properties
Awareness is present at the latest frame
Observation is extending in this direction
Observer is present here (Origin persisted)

The reason the awareness needs to be at the tail is because only then extension happens correctly, the observer at the origin is extending awareness into the depths of the indeterminate expanse. This is explained here are the awareness of the capability to expand from its own at-rest to the subsequent turbulence which is present in the indeterminate. The location at the edge where awareness is located is the "remote individual" to the "observer that is located at the origin". When we live in this world of descriptions, we are at the edge where awareness is present.

This further goes to describe the twin-produced as this:

*vahnyambushikhijA viShNuviDaujaH shacikA dvija*

*sUryAdInAM khagAnAM ca devA jneyAH krameNa ca ||3.18||*

*klIvau dvau saumyasaurI ca yuvatIndubhRgU dvija*

*narAH sheShAshca vijneyA bhAnurbhaumo gurustathA ||3.19||*

*Translates to*

*At the tip of the produced agents of ambu, by the produced equilibrium of fragments, one of the twin-produced is strong due to the gaps of the pattern without stress, appearing to move and also knowing the energies in series. The weak of the two consists of transformations of the pattern and is held by the impulse of the first thought of the twin-produced, the remaining penetration of the individual with the coherent appearance of awakening density exists.*

Here we see the description of the stronger of the twin present at the origin and knowing the energies, because this is the one that is detecting and maintaining the cycle and being affected by the goal of equilibrium, while the weaker of the two which is the awareness which is at the edge knows the "transformation" of those energies and impulses and is the weaker one following the stronger one. Thus

the remote individual is the one that is rendered by description as this consciousness that we know.

**What is an individual?**

While I have been writing saying that "the brain" has been giving us smoothened images, sound, taste, touch etc., experiences, the Sanskrit literature don't refer to it as the brain the way we understand it. It is considered the manas that does this. So, what is the difference between "the brain" and "the manas"?

The ancient literature were more interested in "thoughts" rather than matter. In modern science we consider the neurons of the brain responsible for thoughts. But, if what we have reasoned before is true, i.e., what we observe as reality is just a reflection on the mind's mirror of the truth, we need to ask ourselves, how does the brain come into existence without a sensory perception? In fact, while I have always taken examples that are external to myself in all the above reasoning, we need to ask ourselves "How different is the body from say a leaf or tree that I observe?" If sensory perception is creating a thought, then even my own external and internal appearances, from the organs that form me to the sensory organs that are used to sense, are sensory perceptions or thoughts I have formed of my own body. Or in-fact it is the other way around, I have formed thoughts and hence my body has rendered, which is what appearances and actuals tell me.

Thus, manas is that collection of thoughts that form this "individual as we understand it". It can be anything from emotions, rational, expectations, hopes, wants, ideas, opinions. A collection of all these thoughts is considered to be manas. So, this manas is the one that is giving me these smoothened thoughts. So, "what is an individual?"

In modern science, we assume that we are all individuals first, separate from each other. The stones, the mountains, the forest, the houses, the earth, the sun, the stars, the animals, the trees and so on, we can keep searching and finding more and more individualistic objects that are different from each other. Once we have these individualistic objects, now we try to unify them using an atom, a molecule, a quantum particle, a Higgs field and so on.

But, when we look at it from the perspective given in the Sanskrit literature, we start at the continuous whole. The whole is first in the unified state in which division is established rather than divisions unifying to become a continuous whole again. What does this change in perspective, change of the individual? In this

perspective, the question “What is an individual?” has changed its sense completely. The question rather has become “where is the limit established to create an individual?” We already know that there are “foundation blocks” in the continuous whole based on field of vision of the data. These blocks come together in multitude to form intent and so on. Since it is just a field of vision and not a actual division, it stands to reason that all of these blocks and intents should have still been a single continuous whole. So, why are there individuals?

If we look at Surya Siddanta, it says the individuals are formed because of recognition of “external” and “internal” in the thoughts. So, why does there exist an external and internal? When we stop to think about it, we find that the words that we use to describe, somehow gives us certain pre-conceived notions because of which we do not go the full extent to understanding the actual. “External” and “Internal” for us is always w.r.t space. So, the thoughts come as external to our body and internal to our body. But, when we look at it from the perspective of “data”, it is more a concept of “outlier” and “inlier”.

When we look at a system that is consistently doing a integral of differentials and finding patterns and hence forming knowledge from it, we start realizing that “logic” is formed from data accumulation. At some point, the logic created stops modifying itself according to the data and starts only collecting that which matches its logic that it has already created. Now any data which does not match it’s logic becomes outlier to it and other data belongs to it. This is what I have explained in detail in the book “Surya Siddanta: Emergence of empirical reality”. It says this:

*shanipAtasya bhagaNAH kalpe yamrasartavaH*

*bhagaNAH pUrvamevAtra proktAshrcandroccapAtayoH ||44||*

*ShaNamanUnAM tu sampIDaya kAlaM tatsandhibhiH saha*

*kalpAdisandhinA sArdhaM vaivasvatamanostathA ||45||*

*yugAnAM trighanaM yAtaM tathA kRtayugaM tvidam*

*projbhakya sRShTestataH kAlaM pUrvoktaM divyasaNtavayayA ||46||*

*Translates to*

*Undisturbed appearances existing within series of becomes appears as essentially supported by causal series of becomes beyond which penetration implies appearance of transformed apex. Thus 6*

*thoughts forced by series of changes by momentum of that transition starts to appear as thoughts coming from thoughts existing with transition. Thus 3 stably connected when elapsed making other connected knowledge, circumvents existing increase in produced series of changes beyond which connected illusion exists imperishable.*

Here, if we see the Surya Siddanta says that once an apex has been formed for the path, 6 thoughts are created because of that momentum of the apex, subsequent to which thoughts formed appears to come from the original thought. When there are 3 of such stably connected thoughts, then the rest starts to remain just that which is connected thoughts and becomes external or connected illusion.

Now, we started having such an "external" and "internal" distinction between thoughts, there then form the "individual" with all the thoughts that form the internal and the external becomes the remote to the individual. These set of logically grouped thoughts lead to more secondary, tertiary, fourth level and fifth level thoughts, which together forms the manas or the individual as we understand them.

## The fifth dimension

As I have said before, the manas or thoughts smooths out and renders the underlying workings as this reality around us. A lot of questions arise from such a perspective of reality. But the foremost question that should arise is, "If my thoughts are creating this world around me, then does it imply that the world as I know it, is not the same for all of us?" We need to ask ourselves this question, because there is nothing that is proving to us that all of us create a reality which is the same as each other and same as the animals or plants or any other object in this world around us.

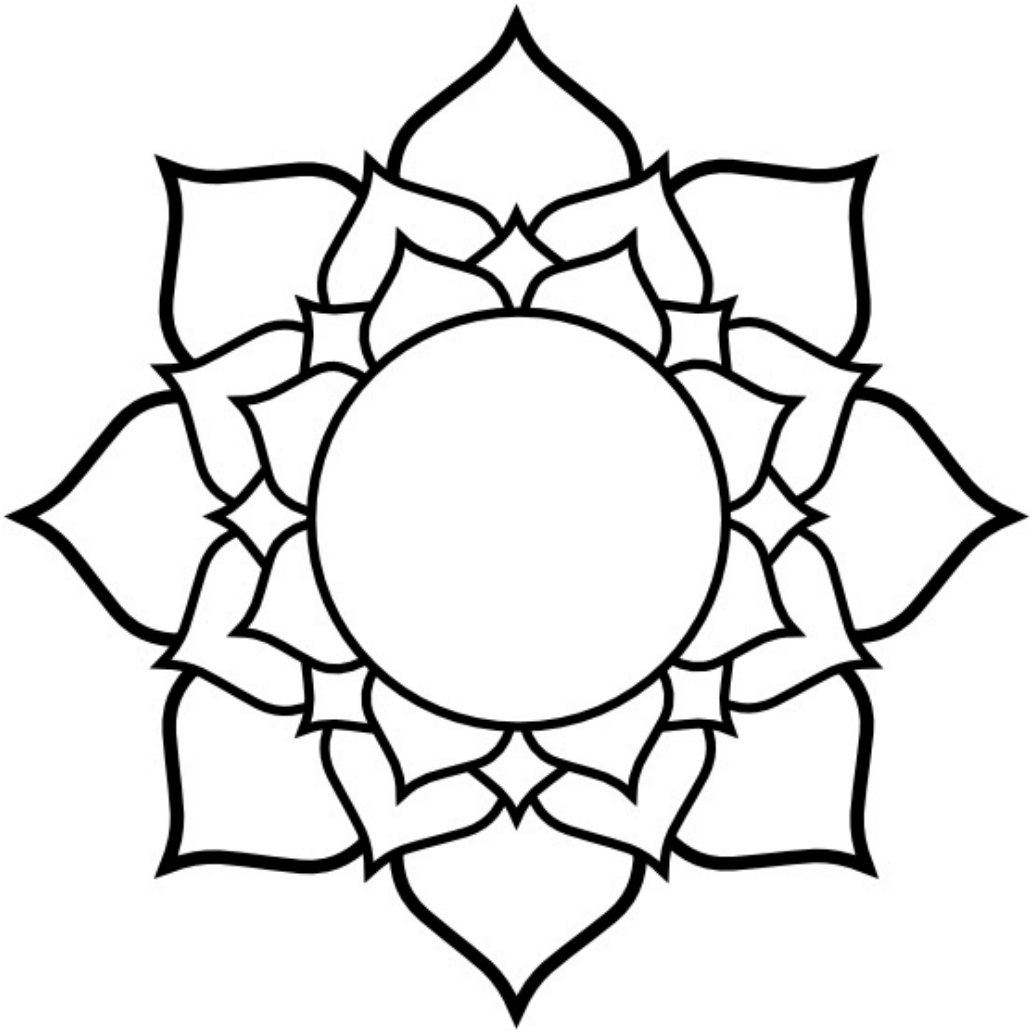
In-fact, we have a lot of proof that each of us see the world differently based on our emotions, thoughts, expectations, hopes and many other related thoughts. We find that the same event is observed differently by different people. Dogs see the world differently obviously because they have the capacity to hear more frequencies, insects such as bees see it much more differently from us because we know they eyesight is much different, plants have a totally different sensory perception. So, if we really think about it, with each of us the sensory perceptions are varied and based on how much of the

sensory perception we are able to glue together to form a world, the world should take on myriad forms.

So, why should we assume that the world is the same for all of us? We are commonly tied together by the continuity, the changes, the cycles and intents as I have explained in the previous chapter. But, how much of that continuity is described and rendered for each individual observable, varies from one observable to the other. This, that is a conclusion based on the described and rendered is called the “fifth dimension” or as Surya Siddanta calls it, “pancadashA”, which is created due to “smara or persistence”. What is worthwhile to note here is, as I have said that each and everyone of us in this reality need not have created the same or similar world with the sensory inputs, it follows that the individuals who wrote the Surya Siddanta or the BHPS also need not have created a similar world as we create into reality around us. This can be a reason why we do not see much of the description related to this layer in the Sanskrit literature.

It should also be noted that while the fifth dimension can be studied, it cannot be described as an abstract same for all observable individuals. Thus studying this dimension needs to be done separately for each individual. While the common elements observed can be studied, there is nothing that governs it. The governing principles are just established as “common sense” as I have explained in the further section. The common sense however is something that can change from one lifecycle to the next lifecycle and need not remain constant across lifecycles.

When we look at evolution from this perspective, i.e., data accumulation to form knowledge, analysis and intelligence, we find that the knowledge present in the prior lifecycles should have been more closer to the process involved in the truth starting to render as this reality around us, while as lifecycles progress, that information is lost to be replaced with more analytical output. Hence, the representation for Brahma or creation is a lotus. It starts at the center closest to the truth and progresses by adding a single petal at a time around this one at a time over and over again till the last layer is completely devoid of the center truth, even thought it is connected to it.



It should be recognized that while the various cycles of the foundation blocks are driven by the intent cycle and changes the work done and introduces work intent to change the truth as I had explained in the previous chapter, the cycle that renders the foundation blocks and the intent cycle are purely descriptive in nature and any changes made in this level of rendering is purely within the rendering. It does not affect the truth or the foundation blocks or the intent cycles. This dimension then, can be seen as an offshoot from the foundation block cycle to provide an appearance and purely a rendering. The Surya Siddanta provides us a description of what in the foundation blocks appears as what in the rendering. But that which is described is purely the very basic common objects. The first that is said about the 5$^{th}$ dimension is as I have listed before in the description of shAnti:

*vakrAyativakrA vikalA mandA mandatarA samA*

*tathA shIghratarA shIghrA grahANAmaShTaghA gatiH ||2.12||*

*tatrAtishIghrA shIghrAkhyA mandA mandatarAsamA*

*RjvIti paNcaghA jneyA yA vakrA sAtivakragA ||2.13||*

*tattadgativashAnnityaM yathA dRuk tulyatAM grahAH*

*prayAnti tat pravakShyAmi sphuTIkaraNamAdarAt ||2.14||*

*Translates to*

*Continuous curve connected to curve, consistently, slowly, very slowly, in this manner very swiftly, acquires eight in conjunction with path. There, very swiftly, in conjunction with obscure, consistently, slowly, very slowly, obtains a motion in the 5th, knowing this, that surely a curve accompanies a curve. Then, that regularly controlled path, if observed in accordance to the acquired, before being established that act of observation, gives the spark of manifestation.*

As the accumulation occurs with intent, continuously, consistently and slowly, the fact that it keeps getting accumulated with every accumulation increases the certainty of the intent. Then that which has formed by the accumulation according to this verse starts getting a motion in our 5th dimension and manifesting in reality. This verse is important to know when an intent will manifest into reality. As we all know, we have many thoughts and intents, but if we do not hold the intent and work towards making it happen, it does not happen and just fades away remaining an intent. Once it has started manifesting, then we have all the other qualities established for the intent in the 5th dimension. Once the spark of manifestation has started, the first quality that needs to be acquired by the intent is direction, which is the next that is said in Surya Siddanta:

*tattvAshrvinoNkAbdhikRtA rUpabhUmidharartavaH*

*khANkAShTau paNcashUnyeshA bANarUpaguNendavaH*

*Translates to*

*The curves in the state of being non-penetrated are rendered indistinct, caused by the nature of becomes construct with gap in 8 curves. In the 5th, the quality of seeking of the non-existent is transformed to directional nature.*

The first to be transformed according to Surya Siddanta is direction. As we saw before, once a logic is established, the patterns that are sought are the patterns that match the logic. Thus, this then becomes the first quality that gets transformed in the 5th dimension as a direction. Again, as I have said, direction is never always space related. Here, the direction indicated is the direction of the intent.

The next to note here is what happens to this spark of manifestation and direction of intent that is established. The Surya Siddanta says, the next is to establish an impulse and supporting points to that impulse:

*shUnyalocanapaNcaikAshCidrarUpamunIndavaH*

*viyaccandrAtidhRtayo guNarandhrAmbarAshrvinaH ||2.18||*

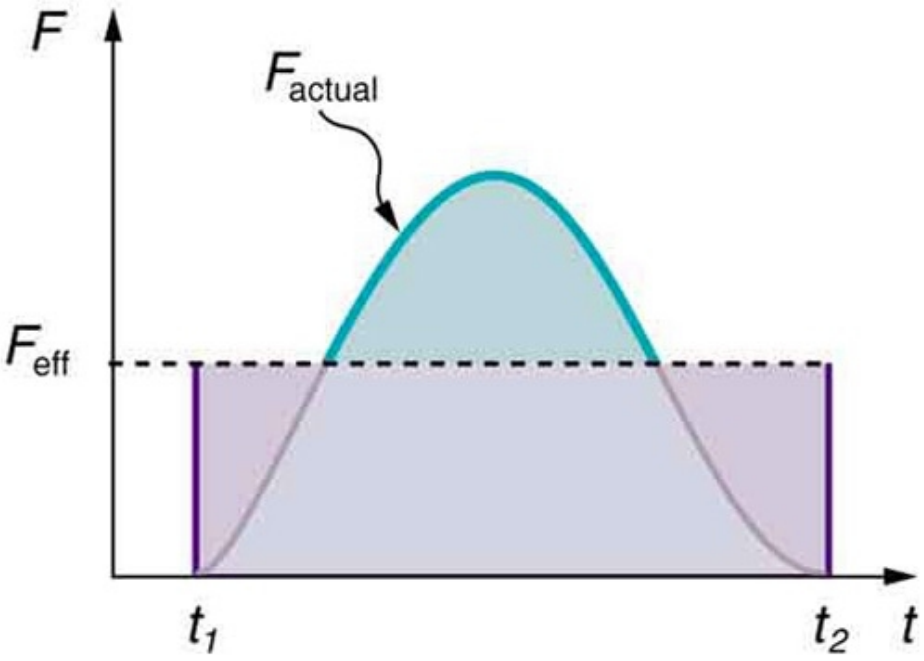
*muniShaDayamanetrANi candrAgnikRtadastrakAH*

*paNcAShTaviShayAkShINi kuNjarAshrvinagAshrvinaH ||2.19||*

*Translates to*

*In the 5th, the non-existent perception of a single penetration whose nature is to pierce, transforms to impulse. When keenly observing quality of the surrounding unpenetrated gap it is transformed to space. The impulse has 6 supported guiding points in the 5th, transformed because of thermal energy with 2(focal points = 2*3 = 6) around the center of axis of the domain of 8, pulsating orbit penetrating the immovable, unpenetrated.*

Again, we need to come away from the trap of pre-conceived notions of words. Impulse for each of us have different meaning. Typically in physics, impulse is an integral of some force and for us takes on the meaning something similar to below:




This is because we have taken time as space frames and have plotted the integral of forces in space. When we take away time from

the picture and look at it as a series of changes rather than time, then we are observing at the same data point, accumulation of differentials which increases the strength of the accumulation as the series increases. Thus, impulse is more the shape of comet, where the intensity is highest at the head which becomes the impulse point, if formed fully, shown as below:

Once the head gets formed, then subsequently multiple can accumulate as it grows. This is the description from BHPS, but when we look at Surya Siddanta, the explanation can be seen similar to the ripples created in water if a drop of water falls on it. Similar to below:

But, whichever description we take, the impulse has to be formed first before the intent can manifest and be maintained in this $5^{th}$ dimension. Subsequently, according to the Surya Siddanta, this impulse needs to grow with persistence, and get transformed into combustibles or thermal agents of the thermal energy and a measure called “gaja” of multitude has to get collected, only then it comes into existence in space. Thus, according to both the BHPS and Surya

Siddanta, everything in this 5th dimension around us starts with intention and is a manifestation of that intention. Subsequently, the Sanskrit literature has left the rendering open, because this is an interpretation of the individual as I have explained before.

**Impact of actions in the rendering on the work intent**

What is important to note here is about "action" in this 5th dimension. While we believe we are acting to change that which is getting rendered, it truly does not control anything in the underlying working. Unless the work intent created in the work intent cycle changes, the underlying driver of the rendering does not change. In fact action is purely a rendering of the movement of the intent cycle rather than action being the driver for the intent of the intent cycle.

This is explained in the Bhagavad Gita verse:

*karmanyevadhikaraste ma phaleshu kadachana*

*ma karma-phalahe-turbhur ma te sangostvakarmani*

*Translates to*

*Always the result is bound to work invoked by the order, advancing the bound result of the order further only binds that accompanying to non-order.*

This simply brings out the correct reason why without changing the work intent nothing can be changed. When a work intent drives the change of the inner workings of reality, that which gets rendered is a result of that change, because it is a description of that result of accumulation of changes. When we change the description, then the only thing that can be changed is that result which has no impact on the change and hence does not re-order the data.

For a comparison in computer parlance, we have all seen web pages that act as the UI for the back end operations. Say, there was a bug in javascript of the web page due to which some data is displayed wrong. We can definitely debug the web page using say chrome developer mode and fix the bug in the javascript as it is run by debugging it. This will display the correct value. But, reloading the page will just display the previous wrong value, because the server has the original javascript that has been reloaded. The javascript in the server has to be corrected to get the correct display.

Similarly here, you can definitely try to change the result in the rendering around us by doing a few actions and this will temporarily also change the rendering. But, it should be noted that it will always go back to the original result of the inner working. This is best noted

when we try to change the growth of plants. Say a sapling has grown in a crooked manner because of the presence of some obstacle that prevented the straight growth of the trunk and it started bending over. Subsequently, we can place a stick or any support and try to straighten the trunk as much as possible, we find that it will just go back to the previous bent trunk when the stick is removed.

But, what is to be noted is that it does not say actions do not introduce work intent. If, we could do an action that introduces a work intent that will introduce a change that triggers accumulation, then action changes the work intent and what is got is not temporary. In the above computer parlance example, if we changed the value displayed in a submittable field and actually submitted the value to the server to change and save itself, then subsequently the values will be the changed value. What is needed is a thorough understanding of what action introduces what work intent, what work intent leads to what accumulation and what accumulation when described will render what we want. Some of these are self-evident and can be done as below.

In the plant example, if we leave the stick as is and make the stick an obstacle for the tree to grow in a certain space, then the plant with automatically grow into a stable shape around the stick over the years. As a tribute to the person who grew these trees below in this shape:

zafojones

Circus Tree: Six individual sycamore trees were shaped, bent, and braided to form this.

The above tree should have been grown with really good care and patience over a period of years, modifying the underlying work

intent, so that the description of the changed accumulations follows a specific rendering rather than just a quick fix.

This can be easily seen in any natural growth or behavior. We can see this when children are growing. They sometimes acquire a bad habit. If you want to break the habit, it requires a lot of patience and constant action to ensure the underlying intent is changed and the behavior is changed. If not they fall back into the habit when something does not go according to what they want. We can see that within ourselves also, certain hardened behaviors such as smoking can only be broken when the intent is highly steady.

**Evolving beyond the fifth dimension**

But, this is not a purely character and natural behavior related theme. This can be seen in technology also. As I have always complained, all our technology just become garbage at the end. So if I take the sum total of all the technology that is created, it is purely garbage, thus the work intent introduced for them is to create garbage. There is no other way to describe it. To explain, let's take some of the most popular technologies. The phone, allows me to communicate to others in the short term. But, the phone itself becomes unusable and has to be thrown in the garbage bin from where it could go to be recycled into other articles or into the sea as a plastic and electronics waste. But, whichever the case, the final outcome for the phone was to become garbage. When I look at it during the time it was in use again, I find it has no outcome to change work intent, unless what I talked using it had any impact on either the person I am talking or on myself. Neither the person I talked to nor I changed inherently because of it. So, the sum total of the existence of the phone was just garbage. Similar is the case with many technologies that we have created. Actually, what we call science and technology has not led to any evolution of humans themselves by changing the work intent that drives the inner working of humans, but just evolution of some of the materials around humans.

So, what do I mean by actual evolution of humans? Let us look at it from the perspective of the people who would have written the ancient literature. As I have indicated before we have no clue what the $5^{th}$ dimension world created by the ancient people should have been. But, definitely what must have happened is that they triggered the creation of a fifth dimension in a certain manner by consistently observing the data in a certain manner and describing them in a certain manner.

What proof I have of this? Obviously the various texts that say what is good and what is bad actions, follow certain rules while acting, etc which is a theme across all religions, clans and every human and has been passed down over and over again over many many generations. If evolution started at the inner workings with observation forming the basic jnanam, parajnanam, Saraswati, Brahma, Lakshmi, Vishnu, then for sure, these have been instrumental in shaping the filtering and storage of that jnanam, vijnanam, guNa in the path so that the fifth dimension analysis and conclusions follow the path that we have followed.

Now, we can either take it to a path where we create garbage of material around us by creating technology the way we are doing now or evolve to the next dimension i.e., the sixth dimension. To do this, we need to ask ourselves what have we done wrong that we are stagnating and not evolving? Let us be clear, *we are stagnating in the area of evolution of human itself*. Neither is the intelligence gained because of the various technologies created nor the technologies themselves are helping us evolve. Infact as many social media apps, comfort oriented gadgets and frivolous endeavors indicate, we are devolving rather than evolving with lesser and lesser ability to reason beyond this visible universe. The ability of a human to think and articulate without the dependance on some external instrument such as the google search has reduced down all the way to near zero, and even with the help the only thoughts that are formed are existing concepts and some trivial modifications to those concepts which are called innovations. All technology has become very addictive to humans that they cannot think beyond and keep them tied down to the technology.

So, why are we stagnating and not evolving? The answer simply is because we have given up the visibility into all the lower levels and are able to observe only the fifth dimension, which is just the dimension of description. So, the sixth dimension has to be there where we would be able to see all the lower dimensions along with the fifth dimension. Only then can we use the fifth dimension conclusions as a guiding principle to direct the work intent of the lower dimensions to be what we want and thus shape reality in that manner rather than via technology which just plays with the result or the rendering and keeps us in delusional delirium of false achievements just because we have accumulated money which is anyways just a bunch of 1’s and 0’s in the current world.

## What is common sense? How is it established?

Next, we should also ask ourselves, while, we are all different from each other, we see a lot of common things similarly. A red car is seen and described as a red car by all of us obviously except when a person is color-blind or simply blind. The hurricane tearing through or the volcano erupting is still described as the same though with various different embellishments by different people. So, what causes this similarity when the world rendered is different between all of us? This brings us to the first question that needs to be answered. What is common sense? How is it established?

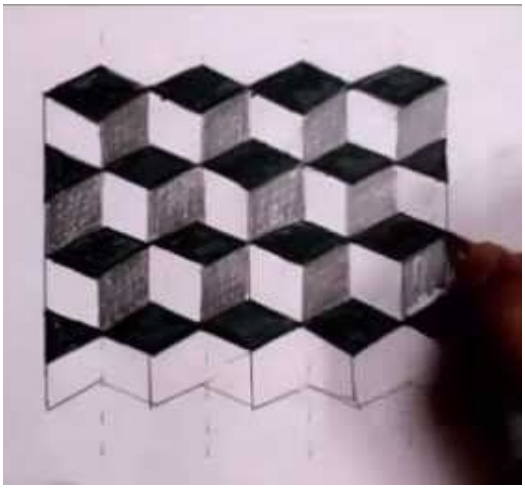
In AI that we have programmed till now, we have been unable to code in "common sense". We have seen this over and over again where a machine is not able to apply the most common sense, if not coded or learnt for, such as a robot vacuum cleaner moving around an obstacle based on the shape and size of the obstacle or a search engine not able to provide the alternatives for say a colgate tartar toothpaste as pepsodent gumcare toothpaste. These need to be coded in as equivalent to be shown. So, what is this common sense that is missing in our intelligence algorithms? This is touted as AGI or artificial general intelligence, but I find it is more than that. It needs a thorough understanding of what we call common sense.

If we look at the descriptions of what is rendered as the $5^{th}$ dimension previous and the formation of intent being a cluster of multitude, we can start getting a glimmer of what common sense really is. It should be that collective intent that is rendered and is present in all individuals.

As I have explained before, in modern science, we first look at the individuals such as atoms, molecules, humans, animals, birds and so on and then go on to unify them into a single explanation because of which we start having problems unifying them. But, the ancients looked at the continuous whole dividing down into individuals, where some logic is present to divide the continuous whole data into individual sets of data. When we look at it from this perspective, we need to start asking ourselves, does this whole continuous get broken down into the trillions of individuals at one go or is there further grouping which then divides down into other groups and so on till an individual called one instance of a human is formed? As I have explained in forces of accumulation, there is a collective pull of various observed patterns to form in a certain manner because of observation and many such observations can come together as coherent knowledge. Which goes to show that they

needs to be sub-groups and more sub-groups of patterns as we keep splitting the continuous whole into individuals.

Thus, we end up with patterns within patterns within patterns and peer patterns and many such combination of patterns which "purify themselves" as the accumulation progresses. As I had given an example in the explanation of "vijnanam", the following:



If we look at it, we have patterns within patterns. A common pattern ties it all together into a coherent whole. So, even though many patterns of black and white and shaded squares squares exist, they all come together to form one whole continuous coherent knowledge. So, if each cube, a combination of one shaded, one white and one black started considering themselves as an individual with all inside it as internal and all other squares outside it as external, then we find we have a common set of patterns that form multiple individual cubes. This common patterns now becomes "common knowledge" for all the cube individuals. This is a collective intent that is present in all cubes.

Similarly, there is a collective intent present in all individuals which is a work intent for common patterns across all humans, or across all dogs or all living beings and so on that get transformed commonly as approximately the same description or rendering giving us "common sense".

So, what is advantage of understanding what common sense is w.r.t to evolving? As we travel through time of human history and observe what has happened, we find that there have been ideas and thoughts that have been introduced that have been adopted as common sense collective by a set of humans and has further progressed to be adopted as a common sense across all humanity.

So, we find that actions that has influence on the work intent need not always be such that it has an impact on only the one individual that influences and creates the work intent. Given that we have taken a continuous whole and divided it to individuals, “work intent” introduced by one individual can also be observed and adopted by other individuals. When this happens, then the common sense of the whole is affected. If this can be done successfully in a controlled fashion, then the whole sub-group can evolve to a common goal of evolution. 21

## Space and time

I believe no explanation of the observable reality around us is complete without explaining space and time. As I have said, time is a concept that is not present in ancient science. What is present is just kAla or series of changes. When these series of changes are traversed by some “self that can observe itself”, it starts off a path of observation of this series of changes by accumulating the differentials of the activated data. The way this is done is using the foundation method, where one division keeps overlaying on the next division and observed. In such a setup, we find that we have three division that play an important role in this whole working. The previous or prior division, the current division and the division that will follow. This is the “become” or the observation that was done when the current division was not present and only the previous division was present, which is the past of our time. With the current division overlayed and we currently experiencing it becomes the present and the “potential for the next division to be overlayed and give us the potential pattern or the work intent” becomes the future. Thus, this past, present and future forms the time that we experience in the $5^{th}$ dimension. Both the Surya Siddanta and the BHPS talk about this. The BHPS has said this about it:

*sarveShu caiva jIveShu sthitaMhyaMshadvayaM kvacit*

*jIvAMsho hyadhikastadvatparamAtmAMshakaH kila ||1.22||*

*sUryAdayo grahAH sarve brahmakAmadviShAdayaH*

*ete cAnye ca bahavaH paramAtmAMshakAdhikAH ||1.23||*

*shaktayashca tathaiteShamadhikAMshAH shriyAdayaH*

*svasvashaktiShu cAnyAsu jneyA jIvAMshakAdhikAH ||1.24||*

*Translates to*

*The entire extracted and the individuals extracted is established in the rear part of the two where awareness is, the individual part of the stable rear and subsequent, both that certainly are parts of the primary nature. By acquiring the unrelenting pattern, the entire pleasant/unpleasant are created unrelentingly, approaches the other and many subsequent parts of the primary nature. The penetration of the capabilities and thus this seeking in the subsequent part attains relentless mixing of the extracted capabilities within themselves and other known thoughts of subsequent individual part.*

Here, the prior part is called the rear part or "hya" or previous and the subsequent is called "adhika". Once this is established, then and only then it goes towards establishing space. But, what is even more important to understand is that "space" according to BHPS or Surya Siddanta is a subsequent transformation of an intent or impulse of the intent and not by itself. The Surya Siddanta says this about space:

*khartuvedA navAdrayarthA diNnAgAstryarthakuNjarAH*

*nagAmbaraviyaccandrA rUpabhUdharashaNkarAH ||2.24||*

*sharArNavahutAshaikA bhujaNgAkShisharendavaH*

*navarUpamahIdhrikA gajaikANkanishAkarAH ||2.25||*

*Translates to*

*The cause of new unlimited gap induced knowledge is the production of directional spiral caused by pulsating orbit in the immovable surrounding space transformation due to the nature of curves penetrating the supporter of becomes construct. The configuration of oscillation is invoked by the transformed configuration of the single penetration of member curvature of the focal point, the new nature of the single unrestrained space curves without multiple is one gaja.*

In the above verses what is described is the gaps in between the activated continuous bits of data that is having an orbit as the traversal path in the indeterminate whole. As we saw before "a gaja is a measure" is important for manifestation into reality. The intent also transforms only when a measure of "gaja" is reached. I will call it as "gaja" since I have no clue what this measure is and I don't believe this measure is transformed into any measurements that we have.

In Surya Siddanta, the concept propounded is that if we take each thought, it is formed of 7 orbits as I had described in the "data

representation" section. These 7 orbits are in a spiral form with each curvature separated by gaps. So, when we are looking at a very basic observer with only capabilities to detect change, they will only see few types of changes. So, a curved gap while is a change, is observed because of the existence of 2 focal points that is indicative of the curve. The focal points are "agents of knowledge" and the gap itself is the knowledge. At the first level many such thoughts come together into a pattern that forms an oscillation of the same pattern which is the thermal energy. Many such pattern oscillations are now connected together with bonds. These come together to form a base network which is the "bhUmi" which is normally translated as "earth". But in my view, it is this network that forms the basis for formation of other individuals, literally translates to "bhU + mi = construct of becomes".

So, when we look at this expanding network having multiple such thought patterns that are oscillating in the pattern of thermal energy, we have a potential to observe unlimited such gaps that form the oscillation. When we look at that gap and transform it in the $5^{th}$ dimension, it becomes the single unrestrained space that we observe around us and is always a multiple of the measure of gaja. In this space then those that traverse that base network form individuals and hence we exist within that space and adopt the transformation of the space and time that is always present till we exist, because they form the basis of our formation.

**Distance and motion as we know it**

For us distance and motion exist only in space. While we have started using distance as a concept in AI for data, it is still in the virtual world. In reality, distance and motion are only space related. From the previous section, we see that space is just a transformation of a base network in which we are traversing governed by the foundation method and appears as such to our thoughts and nothing more. In actual, my speed in the inner workings of reality is limited by the speed of extension of the base network, because I cannot travel faster than the network I am traversing, and I am limited by my speed of penetration of changes in the network which is the one base capability required to traverse the network. In actuality, the actual individual "stays still" at the origin. It is just the awareness of that individual that is extending and expanding in that network. The accumulation till my tip when described by me transforms to this rendering of reality around me. The transformation of interferences of other remote patterns with my pattern at the tip described by me transforms to other remote objects in my reality. So, what is

rendered as distance in space that I experience? And more importantly what do I experience as motion?

Motion and action have been the stumbling blocks for me as I was trying to understand what is being said in the Sanskrit literature about the inner workings of reality. The literature actually does not talk about motion or action much. Many tend to translate kAraka as action and hence think it talks about action. But, as I have said, kAraka must be translated as "work". So, I found everywhere the Sanskrit literature just talked about work and not really action.

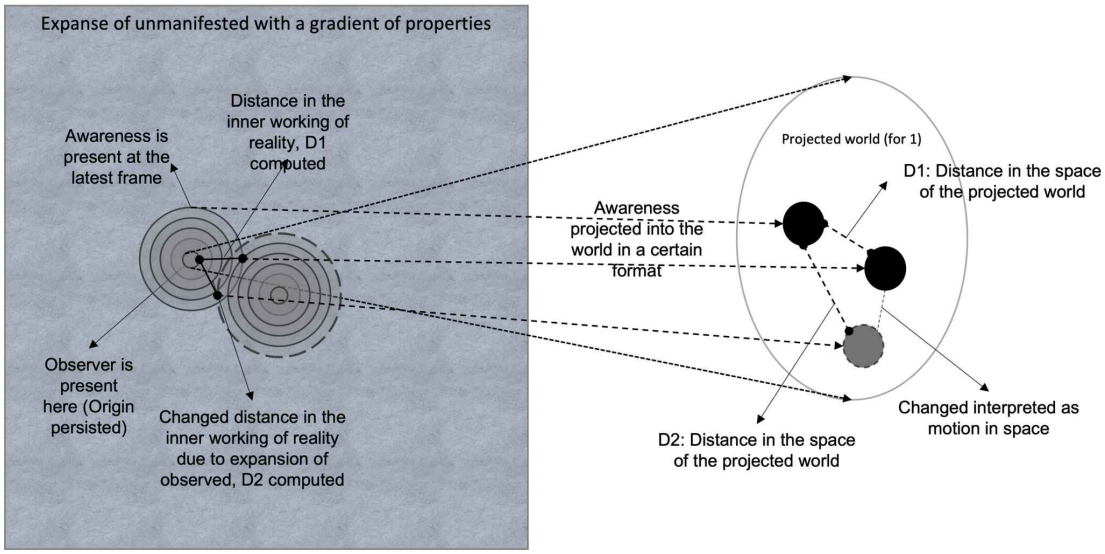
I found that the explanation for the fifth dimension is very less. But, the reason why motion and action have been the stumbling block is because it is a common sense that gets projected as same between multiple individual projected worlds without actually affecting the common base intent. There are only two ways in which I believe this can happen. One way is if these affected the actual work intent and pattern in the inner workings, so that every individual projected the change appropriately in their projected world, as I have explained later.

But, as I have said before, action in this world can happen without affecting the work intent and just an action on the appearance which does not change the underlying order and pattern. This led me to believe that there have to be two types of actions as I have indicated before. Those that affect the work intent which is visible to others, those that don't which is visible only to me in the inner workings. But, we find that there is these actions that is visible to a select few and yet do not have an impact on the underlying truth, which is where a majority of our actions fall. Hence it became a stumbling block.

My take is that the projected world is not just as simple as just projecting the inner workings of reality of just my description. It is a mixed projection of both the inner workings as observed by the inner intent self which I call "internally observed" and the "externally observed" by the sensory perception of the described world which is the external perception which acts on results of the projection of the inner world. The "externally observed" must be causing some "unknown change" in the inner working of the common intent represented by the external objects such as light, mineral, metal etc observables, which we use to achieve action and motion. This is not a direct relation to distance or motion or any change in my inner self, but has some minute change in those observables which has become an agent of knowledge in the common sense and hence is

translated as same between all individuals. Hence, majority of our actions fall within this type of external perception world action projected directly in our worlds without a direct impact on the inner workings.

To really make an impact with motion that changes us, we need to understand how we can move due to change in our work intent rather than just the externally observable motion or action that we do. For this we need to take the concepts of awareness, origin, extension and expansion and map out what could have happened when we see motion in space. The below diagram shows this:


Expanse of unmanifested with a gradient of properties
Distance in the inner working of reality, D1 computed
Awareness is present at the latest frame
Projected world (for 1)
D1: Distance in the space of the projected world
Awareness projected into the world in a certain format
Observer is present here (Origin persisted)
Changed distance in the inner working of reality due to expansion of observed, D2 computed
Changed interpreted as motion in space
D2: Distance in the space of the projected world

For simplicity, I have just used two individuals extending and expanding in perspective 1 which I had used as an example before. I have added the projected world for the first individual (I1). In that projected world exists two individuals I1 and I2 represented by the black circles. I2 is projected because there is an interference of I2's pattern with I1's pattern. Why interference? Because I1 detects patterns and the interference causes a change in pattern and hence creates the work intent expectation of a pattern, thus knowing the existence. This interference is translated and projected in some format. The distance of interference is known in the inner working, computed based on the properties and pattern. I have marked it as a space distance again which need not be same here. It could just be the number of gaps, since gaps translate to space, but it is some detected knowledge agent. This now translates to a distance D1 in the projected world. So, we see that actual distance need not be equivalent to the space distance that we have computed and know so well.

Let's say the the I2 when projected had extended and expanded till the solid lines present for I2. As accumulations progressed, I2 extended and expanded to the dotted circle. Now, the interference pattern has changed. So there is a recomputation based on the pattern observed in the inner workings and a distance D2 computed in the projected world space. There is a difference and hence I2 in the projected world is rendered as the dimmed circle by I1, giving the illusion of motion in space. In reality all that has occurred is that I2 has extended, expanded or done both to its awareness.

A similar concept can be applied to I1 for its own motion. The projection of I1 in the projected world is again just of the awareness or the tip. When this changes without expanding its awareness, it should be projected at the same location in the projected world. Move the awareness in any direction, either by expanding or just a change in the focus of the direction where the awareness is present in the circle I have drawn and there is a self-motion in the projected world.

If we look at it from this perspective, we find that we may actually have to expand less, but be able cover a bigger distance in the transformed world. We did not have to modify all the elements around us just to create the appearance of motion as we do right now and generate so much of garbage and be limited by unnecessary limitations of the projected world. And it is quite possible that we could have evolved to the sixth dimension.

**A different view of scientific experimentation (energy, work, flow & fluidity)**

As I have been writing in all my blogs and books, I find that modern science is like a user's manual for reality rather than the developer's manual where we know exactly how the whole reality is working. When we look at it from this perspective, I find that most of our scientific concepts, theorems, experiments etc are just on the description or rendering and does not take the inner workings into consideration. We have limited ourselves to the science that is described in space and time continuum and introduced languages such as mathematics to express these descriptions for communication.

I find that continuing to stay just in space and time will not help us answer any question. If you have been following what I have been explaining, it is only going to introduce more observations which activates more and more data to satisfy the observer's imagination to become a complete confirming observation satisfying

all the common sense parameters, to ensure that the results of the experiments also confirm to the imagination. Which imagination gets a confirmation and becomes common sense and which does not is going to depend on the strength of the intent formed by the observer. This being reinforced by the multitude the observer can gather onto their side. This has been seen over and over again where a number of scientific concepts now-a-days are proven by statistics of inadequate sample population or popularity of acceptance of the concept. Given we now also have social media, this has become even more useful to introduce more and more imaginative concepts. We need to stop trying to build more imaginative layers on already shaky imaginations and understand the difference between descriptions and actuals and try to find a science that studies the underlying truth rather than the description.

If we read what is really given in the ancient Sanskrit literature, my personal view is that we should start finding inner working explanations when we experiment as I have described in the forces of accumulation section. We need to start looking at experiments and explain them in terms of energy, potential to exist as a pattern, the fluidity of the pattern and the flow of the pattern that will form the scientific concept and find if there is an explanation or is the experiment and concept proposed just my imagination.

# Appendix: Can we implement an AI algorithm using these concepts?

All through the book I have explained where possible with possible implementations that can help the existing use case. But, what needs to ask ourselves is “Can we implement an AI algorithm or a knowledge and intelligence based system with just the computers that we have”? I find it is very difficult to do so. The primary requirement of data representation seems to be a very sticky problem to solve. Just turning data to an analog data representation does not solve the problem. The next requirement is of a realizing logic from the data. So, how do we modify the processor from the computers from what it is currently, a hard coded logic to a flexible logic processor. What is more important for us to analyze is can the technology that we have currently be utilized in some manner to create a system that can help implement the knowledge and intelligent system? I find the answer to this is a resounding no. To explain it further.

## Trying to solve the problem of data representation

Trying to create a system by tying together the particles or any material modifications that we know today to achieve the data representation that we want for a AI system that can implement the algorithm is not possible. We do not know of anything with the continuity nor the properties required for implementing potentials or active or inactive data. We do not also know how we can get the particles with material properties to have interference to form a new data that can be inturn a correct data that is relevant and related to the original data that formed it. Data representation is the major and the first hurdle that needs to be crossed to create an effective AI system.

We can use light to represent data, but the modulations of light that we use is tied deeply to the way we represent data i.e., as digital representation. The first concept to break is digital data and to create analog data. The next to break is the mathematical representation of the values, rather than seeing data as individual values such as ‘1’, ‘2’ or ‘1.0980’ and so on, we need to see them as continuous patterns that can be changed and have a series of numbers that can be got if required by the algorithm.

The only concept that we have which can act this way is energy. Given we are anyway a thermal system, we might as well use thermal energy to implement data. Heat energy can be used to represent data. But, the sensors that we have are point based sensors such as a thermometer rather than area sensors such the thermal imaging devices. Thermal imaging devices are not sensors, they just create images. Moreover the sensitivity of our sensors to measure a pattern in energy modulations is not present.

We need to create area thermal sensors that create a map of the heat around an area that the sensor is placed in. Something like a heat sink giving an output over the area it has absorbed the heat which can then be used. But, given that technology that we have currently can only do a very small area measurement and it can only give us pointed values, this also does not work. The major problem is that all our sensors are electronic in nature. The problem with electronics is the way we have defined and measured current and voltage. These are measured and used as single values. While in true materials electrons flow along a area, we consolidate them into a single wire and try to measure it in that manner. Hence, irrespective of what the material is capable of, we only get point based sensors and values.

Silicon based components are also just used as binary with a single flow or not flow concept. Even though dopants is a concept in semiconductors and has the capability to have an area flow of electrons as we may find in learning algorithms, we don't use it for anything else other than a control of single flow of electrons. So, a component such as a diode can be used to allow only a single direction flow or a transistor can be used as a regulator, an amplifier etc which is again just a control of a single flow of electrons and not in an area. Thus, even in an application such as a digital camera where an area control is required, we have split the silicon down into pixels and measured the electron flow at each pixel and do not have a way to measure the inherent area electron flow in the material itself.

What we truly need is some sort of material whose molecular bonds are heat sensitive and either form or break based on the intensity of heat present. Thus, if such a material can be passed over an area, then a measurement of the types of molecular structures formed gives us the information on the intensity of heat in the area. This can then be used to drive the next step as activated data.

## A learnable as a processor rather than a pre-defined instruction set processing

As I have said before, our current computer systems are driven by logic. The processor has more prominence than the storage of data. Systems can be built even without any form of data storage, purely on logic with the registers holding and modifying temporary data. While logic acts on data, creates or modifies data, logic is processed and hence data changes. Thus, when the AI algorithms are developed, they focus more on algorithms or logic to learn rather than the data that have the intelligence in them. The data is incidental acting as input to either train the algorithm or act as input to drive the logic to a predictable conclusion. This does not help us go further.

When we look at the way reality works, we find that the logic is learnt from the data by organizing the data in some natural order and hence realizing a logic. The problem with our hardware or specifically the processor is that we have pre-arranged the various components into a static circuit or rather an offline designed circuit. Subsequently, we are trying to learn an algorithm in the software that is processed by this circuit. This will always only lead to a pre-defined logic that is supported by the processor.

We need an environment in which nothing is pre-coded and everything can change as the learning occurs. What is provided is just the underlying abstract logical frameworks that allow a change as more data is pumped into the system. Everything else other than the underlying flow of electrons and hence current and variation of voltage are just derived parameters which are restricted by these basics of the circuit.

The best system that we can build is one in which the circuit itself is learnt as the data is passed through or over it. Then the learnt circuit becomes the AI which can be provided all the other appearances. So, how can we create a system that can learn a circuit?